Critical Release Notice

Publication number: 297-2621-371
Publication release: Standard 05.04

The content of this customer NTP supports the SN07 (DMS) software release.

Bookmarks used in this NTP highlight the changes between the baseline NTP and the current release. The bookmarks provided are color-coded to identify release-specific content changes. NTP volumes that do not contain bookmarks indicate that the baseline NTP remains unchanged and is valid for the current release.

Bookmark Color Legend

Black: Applies to new or modified content for the baseline NTP that is valid through the current release.

Red: Applies to new or modified content for NA017 that is valid through the current release.

Blue: Applies to new or modified content for NA018 (SN05 DMS) that is valid through the current release.

Green: Applies to new or modified content for SN06 (DMS) that is valid through the current release.

Purple: Applies to new or modified content for SN07 (DMS) that is valid through the current release.

Attention!

Adobe Acrobat Reader 5.0 or higher is required to view bookmarks in color.

Publication History

September 2004

Standard release 05.04 for software release SN07 (DMS).

Added additional NetworkBuilder-related data schema information to the CAINPARM table to address CR Q00816405.

March 2004

Standard release 05.03 for software release SN06 (DMS).

Change of phone number from 1-800-684-2273 to 1-877-662-5669, Option 4 + 1.

297-2621-371

Digital Switching Systems

UCS DMS-250

Local Number Portability (LNP) Application Guide

UCS12 Standard 05.02 November 1999

NORTEL NETWORKS

How the world shares ideas.

Digital Switching Systems

UCS DMS-250

Local Number Portability (LNP) Application Guide

Publication number: 297-2621-371
Product release: UCS12
Document release: Standard 05.02
Date: November 1999

Publication history

November 1999

Standard release 05.02 for UCS12 software release.

Added a CIC routing section to Chapter 3.

August 1999

Preliminary release 05.01 for UCS12 software release.

Removed the EOPS information from Chapter 3, “Feature interactions”.

> **ATTENTION**
> The UCS12 software release does not support Enhanced Operator Position System (EOPS) functionality. The UCS software continues to support operator-assisted calls through other platforms such as Enhanced Services Provider (ESP). Refer to Appendix A in the *UCS DMS-250 Feature Change Reference Guide* for additional information about EOPS removal.

Removed CDR field, RTELIST, from Table 2-8. UCS12 software release does not support this field. This field still applies to the UCS07 software release.

Changed title of Chapter 4 from “SS7 and TCAP messaging” to “SS7 ISUP and SS7 TCAP messaging.”

Removed EOPSTST from the “CDRTST and EOPSTST” section of Chapter 8, “NetworkBuilder tools.”

Added data for the LNP_4IMT_RLT CAINPARM to Table 2-6.

Added SS7 Intra-IMT to Figure 2-2.

Added a section for the feature interaction of the LNP_FOR_RX_SELECTOR and the LNP_4IMT_RLT CAINPARMs to Chapter 3.

May 1999

Standard release 04.02 for software release UCS11.

March 1999

Preliminary release 04.01 for software release UCS11.

Changes made to this document include

- added the ***O_Abandon*** EDP
- added the *Switch_Hook_Flash* event at the ***O_Mid_Call*** EDP
- added the CAIN907 log, “Unarmed EDP Action”
- added the OABANDON and SHF tuples to the CAINTRIG OM group
- added the ALTBILL and LATA CDR fields
- added the ***AMAAlternateBillingNumber***, ***AMALineNumber*** parameters to the **Analyze_Route**, **Continue**, and **Disconnect** messages

November 1998

Standard release 03.02 for software release UCS09.

October 1998

Preliminary release 03.01 for software release UCS09.

Contents

About this document

This document covers the role of the UCS DMS-250 switch in providing Local Number Portability (LNP) and the capabilities of the NetworkBuilder product for off-board communication. Understanding, planning, datafilling, and testing LNP on the UCS DMS-250 switch is covered.

Intended Audience

This publication is intended for telecommunications engineers, technicians, switching system developers, operating company personnel, or anyone requiring technical information on UCS DMS-250 LNP functionality.

This document assumes the user's switch is installed, commissioned, and active.

Personnel implementing this feature require the following:

- Table Editor training
- Nortel Networks approved datafill, translations, and maintenance training

How this document is organized

The chapters in this document provide the following information:

Chapter 1, "LNP overview"

Chapter 1 provides a functional overview of LNP on the UCS DMS-250 switch.

Chapter 2, "Provisioning NetworkBuilder for LNP"

Chapter 2 provides instructions for provisioning the switch with NetworkBuilder for LNP.

Chapter 3, "Feature Interactions"

Chapter 3 provides information interactions between LNP and other features.

Chapter 4, "SS7 ISUP and SS7 TCAP messaging"

Chapter 4 provides information on agent setup (Signaling System 7 [SS7] ISDN User Part (ISUP) and Transaction Capabilities Applications Part [TCAP]) message protocols used to provide LNP services.

Chapter 5, "Call processing"

Chapter 5 provides information on LNP call processing.

Chapter 6, "Outgoing message parameters"

Chapter 6 provides information on the population of LNP data in outgoing message parameters.

Chapter 7, "Incoming message parameters"

Chapter 7 provides information on the population of LNP data in SCP response parameters.

Chapter 8, "NetworkBuilder tools"

Chapter 8 provides information on LNP interactions with NetworkBuilder tools.

How to check the version and issue of this document

The version and issue of the document are indicated by numbers, for example, 01.01.

The first two digits indicate the version. The version number increases each time the document is updated to support a new software release. For example, the first release of a document is 01.01. In the *next* software release cycle, the first release of the same document is 02.01.

The second two digits indicate the issue. The issue number increases each time the document is revised but rereleased in the *same* software release cycle. For example, the second release of a document in the same software release cycle is 01.02.

This document is written for all UCS DMS-250 offices. More than one version of this document may exist. To determine whether you have the latest version of this document and how documentation for your product is organized, check the release information in *UCS DMS-250 Master Index of Publications*.

References in this document

The following documents are referred to in this document:

- *Advanced Intelligent Network (AIN) 0.1 Switch—Service Control Point (SCP) Application Protocol Interface Generic Requirements*, TR-NWT-001285
- *DMS-100 Family Software Optionality Control User's Manual*, 297-8991-901
- *UCS DMS-250 Billing Records Application Guide*, 297-2621-395
- *UCS DMS-250 CAIN/FlexDial Interactions*, 297-2621-372
- *UCS DMS-250 Commands Reference Manual*, 297-2621-819
- *UCS DMS-250 Data Schema Reference Manual*, 297-2621-851
- *UCS DMS-250 Feature Change Reference Guide*, 297-2621-050
- *UCS DMS-250 FlexDial Framework Application Guide*, 297-2621-390
- *UCS DMS-250 International Application Guide*, 297-2621-327
- *UCS DMS-250 Logs Reference Manual*, 297-2621-840
- *UCS DMS-250 Master Index of Publications*, 297-2621-001
- *UCS DMS-250 NetworkBuilder AIN 0.2 TCAP Protocol Definition*

  Note: For more information on how to obtain *UCS DMS-250 NetworkBuilder AIN 0.2 TCAP Protocol Definition* contact your Nortel Networks representative.

- *UCS DMS-250 NetworkBuilder Application Guide*, 297-2621-370
- *UCS DMS-250 Operational Measurements Reference Manual*, 297-2621-814
- *UCS DMS-250 Programmable Service Node (PSN) Application Guide*, 297-2621-380
- *UCS DMS-250 Software Optionality Control (SOC) User's Manual*, 297-2621-301
- *UCS DMS-250 SS7 RLT Feature Application Guide*, 297-2621-345

What precautionary messages mean

The types of precautionary messages used in Nortel Networks documents include attention boxes and danger, warning, and caution messages.

An attention box identifies information that is necessary for the proper performance of a procedure or task or the correct interpretation of information or data. Danger, warning, and caution messages indicate possible risks.

Examples of the precautionary messages follow.

ATTENTION Information needed to perform a task

ATTENTION

If the unused DS-3 ports are not deprovisioned before a DS-1/VT Mapper is installed, the DS-1 traffic will not be carried through the DS-1/VT Mapper, even though the DS-1/VT Mapper is properly provisioned.

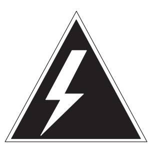
DANGER Possibility of personal injury

DANGER

Risk of electrocution

Do not open the front panel of the inverter unless fuses F1, F2, and F3 have been removed. The inverter contains high-voltage lines. Until the fuses are removed, the high-voltage lines are active, and you risk being electrocuted.

WARNING Possibility of equipment damage



WARNING

Damage to the backplane connector pins

Align the card before seating it, to avoid bending the backplane connector pins. Use light thumb pressure to align the card with the connectors. Next, use the levers on the card to seat the card into the connectors.

CAUTION Possibility of service interruption or degradation

CAUTION

Possible loss of service

Before continuing, confirm that you are removing the card from the inactive unit of the peripheral module. Subscriber service will be lost if you remove a card from the active unit.

How commands, parameters, and responses are represented

This document conforms to the following conventions.

PICs, TDPs, EDPs, triggers and events

PICs are represented in Helvetica Bold font, for example:

O_Null

TDPs and EDPs are represented in Helvetica Bold-Italic font, for example:

Origination_Attempt

Triggers and events are represented in Helvetica Italic font, for example:

Off_Hook_Immediate

Messaging

Messages are represented in Courier Bold font, for example:

`Info_Analyzed`

Parameters are represented in Courier Bold-Italic font, for example:

`CalledPartyID`

Extension parameters and any sub-parameters are represented in Courier font, for example:

`t1Overflow`

Input prompt (>)

An input prompt (>) indicates that the information that follows is a command:

>BSY

Commands and fixed parameters

Commands and fixed parameters that are entered at a MAP terminal are shown in uppercase letters:

>BSY CTRL

Variables

Variables are shown in lowercase letters:

>BSY CTRL ctrl_no

The letters or numbers that the variable represents must be entered. Each variable is explained in a list that follows the command string.

Responses

Responses correspond to the MAP display and are shown in a different type:

```
FP 3 Busy CTRL 0: Command request has been submitted.
FP 3 Busy CTRL 0: Command passed.
```

The following excerpt from a procedure shows the command syntax used in this document:

1 Manually busy the CTRL on the inactive plane by typing

>BSY CTRL ctrl_no
and pressing the Enter key.

where

ctrl_no is the number of the CTRL (0 or 1)

Example of a MAP response:

```
FP 3 Busy CTRL 0: Command request has been submitted.
FP 3 Busy CTRL 0: Command passed.
```

Local Number Portability Overview

Local Number Portability (LNP) provides three types of services: Service Provider Portability, Service Portability, and Location Portability. This document discusses Service Provider Portability only.

Service Provider Portability allows subscribers to retain their directory number (DN) when moving among different service providers.

With Service Provider Portability, neither the local exchange carrier (LEC) nor any other carrier may be identified by the NPA-NXX portion (the office code) of an offnet number. The office code does not identify the end-office switch serving the DN. The DN may reside on a switch whose office code is different than the dialed NXX.

As soon as the first DN in an office code is ported to another central office switch, that office code is then considered “open” to number portability, and any calls to a number with that office code need to be checked to determine if they are ported numbers. For the purpose of this document, Service Provider Portability is referred to as Local Number Portability, or LNP.

Role of the UCS DMS-250 in LNP

Simply stated, the UCS DMS-250 takes an incoming call, determines if it is a ported number, and routes it to the appropriate destination. This involves using the capabilities of the NetworkBuilder application for off-board communication with a service control point (SCP).

NetworkBuilder support for LNP can be organized into four main areas:

- Support for LNP information returning from an SCP on a NetworkBuilder query. NetworkBuilder provides the capability to process LNP information that may return in a CAIN **`Analyze_Route`** response message or in a Bellcore *TR-NWT-000533* IN/1 **`Connect`** response message.
- Support for tandeming LNP information. If the UCS DMS-250 switch receives incoming LNP information, the switch is required to provide the ability to tandem it across. This capability is supported in UCS IEC02 loads and beyond.

- Support for LNP determination. When DNs have been ported in an end-office, then an LNP determination must be made. NetworkBuilder uses an LNP technique called the Location Routing Number (LRN) method which utilizes an in-switch trigger table, OFFCCODE, to perform address screening on origination. This table contains the offices codes or NPAs which are considered “open” to LNP, because one or more DNs with those office codes or NPAs have already been ported. If a call is subscribed to NetworkBuilder through the OFFCCODE trigger and the subscriber number is a 10-digit offnet national number, then the switch has the ability to check the OFFCCODE table to determine what action needs to be taken based on the given criteria. This action could include querying the LNP SCP to determine if the subscriber has changed carriers (ported). If so, the returned message would contain routing information, enabling the call to be routed to its destination.
- Support for home location routing number (LRN) processing. The UCS DMS-250 can be assigned, through table TERMLRN, up to 16 unique LRNs. These LRNs are referred to as the switch’s home LRNs. When a switch receives an LRN, it determines whether the LRN is a home LRN, then routes the call based on the information the switch receives from the LNP SCP.

For a description of the NetworkBuilder product, refer to *UCS DMS-250 NetworkBuilder Application Guide*.

LNP terminology

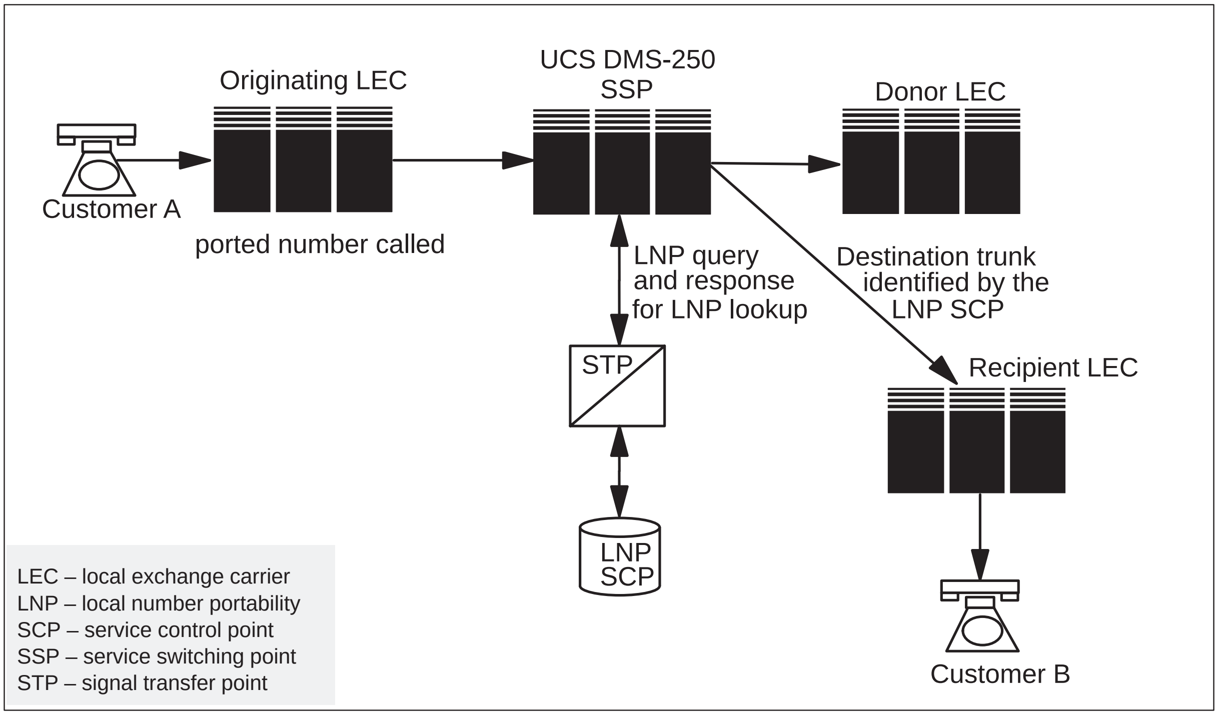
The following terms are related to the LNP network model:

- Originating Switch – the switch from which a call originates.
- Donor Switch – the switch from which a number was ported. The switch used to service the end user before Service Provider Portability.
- Intermediate switch – the UCS DMS-250 switch an LNP call encounters, but to which the call may not terminate. The intermediate UCS DMS-250 switch can initiate Home LRN processing, NetworkBuilder queries, LNP queries, or can tandem information.
- Recipient Switch – the switch to which a number has been ported. The switch which now services the end user.
- LRN – location routing number. The LRN is a 10-digit number that uniquely identifies an end-office to the network for call routing purposes.
- LRN method – location routing number (LRN) method. The LRN method is a technique used to query an LNP SCP when a call is originated. The tandem (UCS DMS-250) switch receives an LNP SCP response message and routes the call based on the information returned.
- SSP – service switching point (UCS DMS-250)

- SCP – service control point
- LNP SCP – LNP service control point. An LNP SCP is an SCP that uses the LNP global title value (SSN), supports AIN 0.1, and functions as described in this document.

Figure 1-1 shows an example of an LNP call.

Figure 1-1
Example of an LNP call



What is NetworkBuilder?

NetworkBuilder provides optional intelligent networking (IN/1) and optional advanced intelligent networking (AIN) software to the UCS DMS-250 switching platform. The IN/1 system NetworkBuilder supports is based on Bellcore's *TR-NWT-000533* specifications. The AIN system is called Carrier AIN (CAIN). CAIN is based on Bellcore's AIN 0.2 specifications.

General AIN

AIN 0.2 allows call processing to be off-loaded from the SSP to a customer-defined SCP. The AIN SCP is able to take control of a call and direct call processing. The intelligent SCP may also contain feature logic and necessary databases.

AIN network model

Software is developed on AIN peripherals to interact with AIN software on the SSP. The AIN network may consist of the following AIN peripherals:

- service switching point (SSP)
- service control point (SCP)
- signal transfer point (STP)
- service management system (SMS)
- service creation environment (SCE)
- intelligent peripheral (IP)

An SSP and an SCP are required for an AIN network. Peripheral software development is independent of SSP software development.

Note 1: CAIN requires the use of a UCS DMS-250 switch as the SSP.

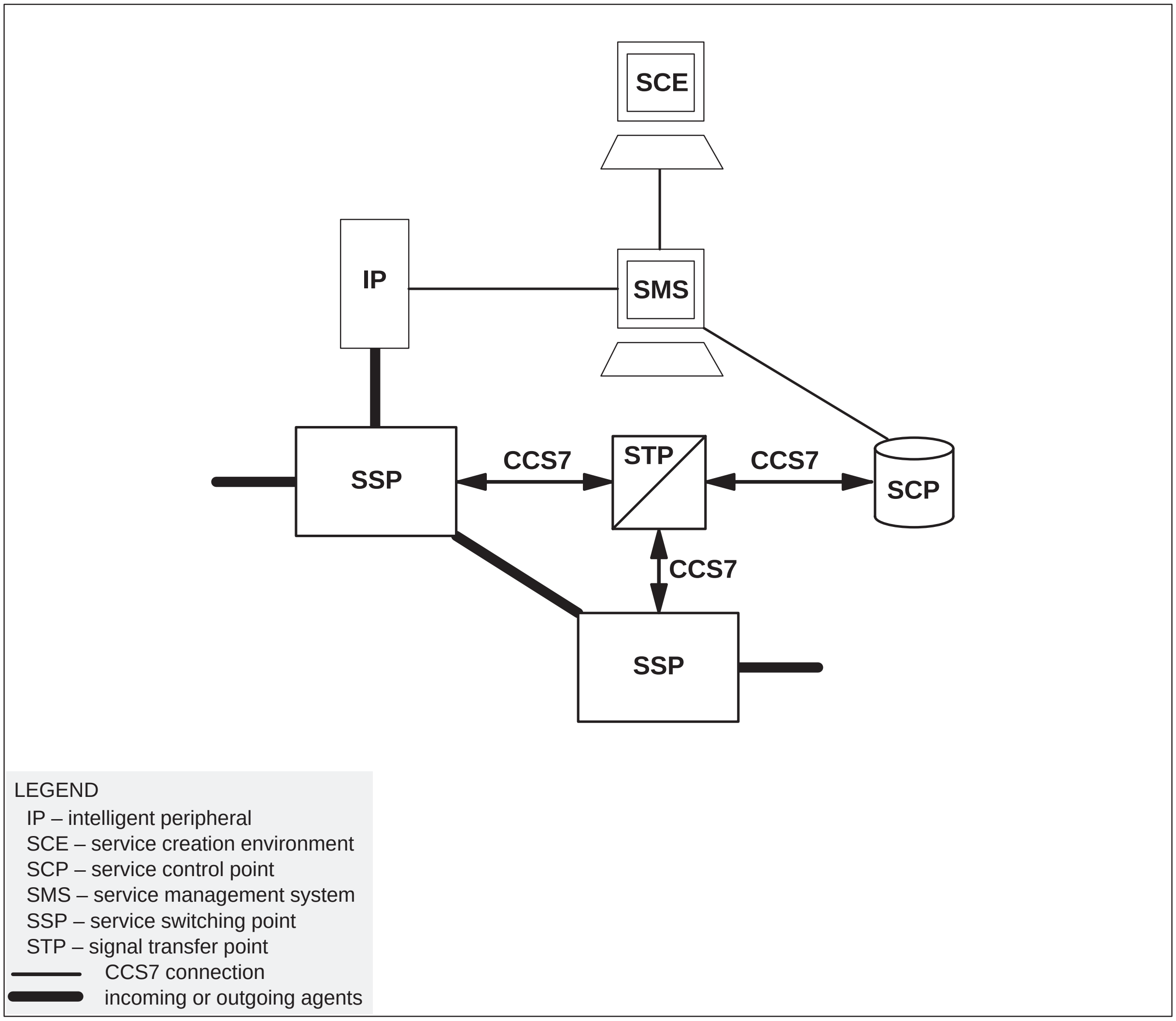
Note 2: Nortel Networks also offers an SCP product called ServiceBuilder.

Bellcore specifications

Bellcore specification *TR-NWT-000533* defines the IN/1 functionality supported by NetworkBuilder. *GR-1298-CORE* and *GR-1299-CORE* define AIN 0.2 supported by NetworkBuilder. However, the Bellcore specifications define local exchange carrier (LEC) functions that don’t necessarily apply to interexchange carriers (IEC). Also, the Bellcore specifications don’t meet the needs of unique IEC applications.

Figure 1-2 shows the hardware components available for designing an AIN system.

Figure 1-2
AIN network model



NetworkBuilder call model

The NetworkBuilder call model defines key states (known as points in a call [PIC]) within a call where processing may be off-loaded to the SCP. SCP interaction at any specified PIC provides greater control of the call services.

At each PIC, call processing encounters trigger detection points (TDPs) and examines these triggers. NetworkBuilder uses the **Analyze_Information** PIC and ***Info_Analyzed*** TDP for LNP triggering.

When call processing enters the **Analyze_Information** PIC and encounters the ***Info_Analyzed*** TDP, in-switch logic and LNP-specific datafill are consulted in table OFFCCODE to determine if the trigger criteria is met. When the criteria is met, the switch can off-load call processing to the SCP. The call may continue in-switch processing, or call processing may be suspended and direction requested from the SCP.

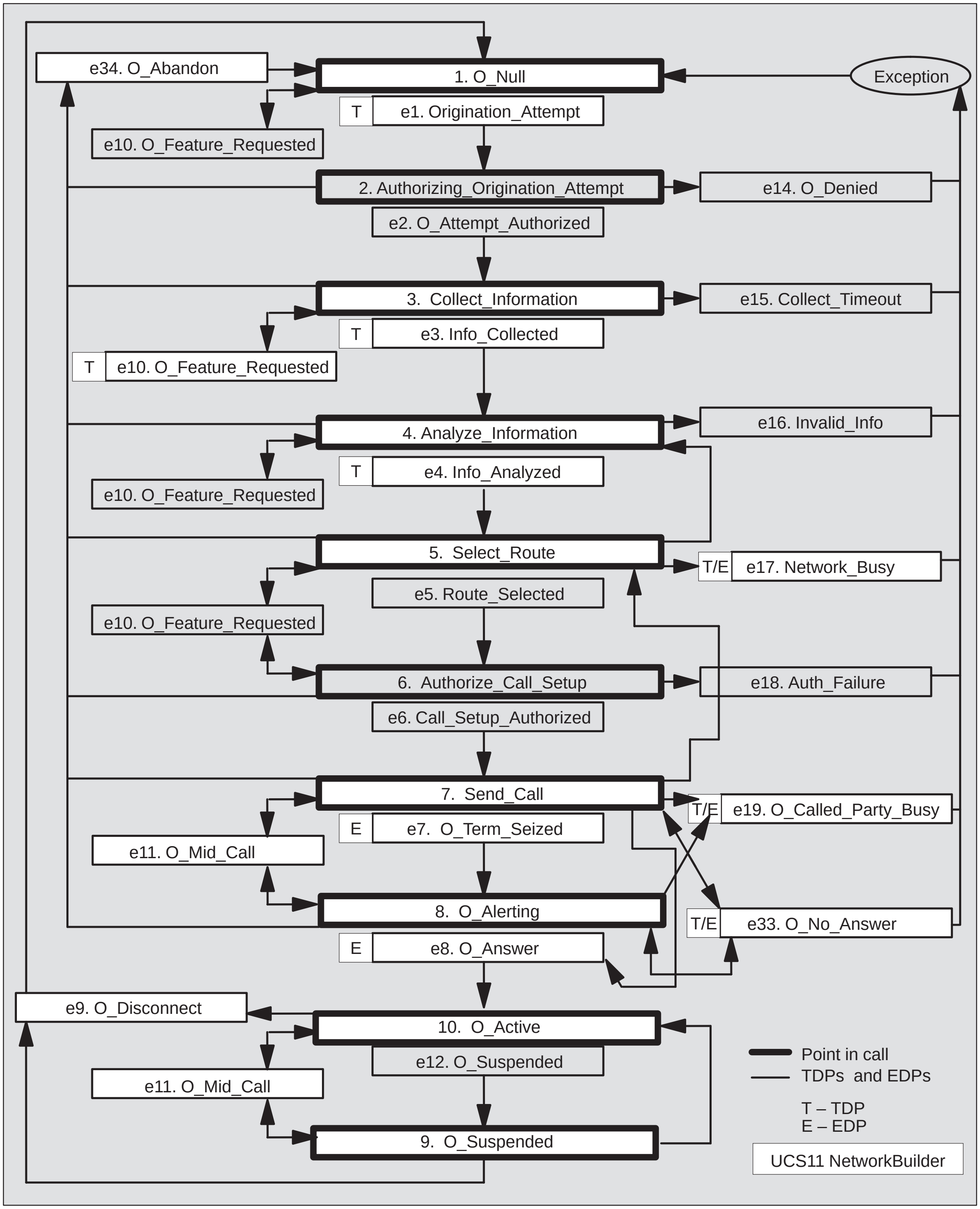
NetworkBuilder also supports event detection points (EDP) at several PICs. EDPs are similar to TDPs. When a call queries the SCP, the SCP may return a conversational package which includes a call-related component such as an **`Analyze_Route`**, **`Continue`**, or **`Collect_Information`** message and a non-call related **`Request_Report_BCM_Event`** component containing a list of one or more EDPs that occur later in the call model. The list is used to activate or “arm” EDPs. EDP “arming” indicates that the SCP has informed the switch to send an EDP-Request or EDP-Notification message back to the SCP when the EDP is encountered. The list of EDPs to arm is called the next event list (NEL). After the call-related component is processed the call follows its standard logic, ignoring all TDPs and triggers, until the armed EDP is encountered or EDPs are deactivated.

Because LNP queries are performed by encountering the ***Info_Analyzed*** TDP and consulting trigger criteria in table OFFCCODE, it is possible to encounter the ***Info_Analyzed*** TDP for *Office_Code* triggering while EDPs are active. No other TDPs or triggers are considered while EDPs are active.

Note: A transaction that sends an LNP query and receives a non-call related **`Request_Report_BCM_Event`** component is closed, but LNP processing of the call-related component continues.

Figure 1-3 shows the NetworkBuilder originating call model.

Figure 1-3
NetworkBuilder originating call model



NetworkBuilder subscription

Any call originating on an LNP-supported agency call can also subscribe to LNP services:

- dedicated access line (DAL)
- FGB
- FGD
- Signaling System 7 (SS7) Inter-machine trunk (IMT)
- AXXESS
- primary rate interface (PRI)
- SS7 Intra-IMT used in a release link trunk (RLT)

Although support for originating calls is restricted to the listed agents, all NetworkBuilder-supported terminating agencies are supported.

NetworkBuilder call processing stores up to six subscription methods (groups) for use throughout the call. The six groups are determined by the following means:

1 SCP-returned CAIN group
2 Address subscription
3 Authorization code subscription
4 ANI subscription
5 Agent subscription
6 Office subscription

Generally LNP services are subscribed on an office basis through the CAIN_OFFICE_GROUP parameter in table CAINPARM. This parameter is datafilled to indicate the CAIN group to be used for trigger evaluation. The CAIN group must already exist in table CAINGRP. The CAIN group, in turn, enables the *Office_Code* trigger. Enabling the *Office_Code* trigger provides an index into the OFFCCODE table which contains the call conditions required to query the LNP SCP and the actions the switch takes when the conditions are met.

The UCS DMS-250 switch can receive an originating call from one of the following subscribing to a CAIN group:

- a LEC through an AXXESS, FGB, or FGD agent
- another UCS DMS-250 or equivalent switch through an SS7 Inter-IMT agent
- a PBX that is linked directly to the switch by a DAL or PRI agent

When the UCS DMS-250 switch receives an originating call from one of the above and the agent subscribes to a CAIN group that enables the *Office_Code* trigger, NetworkBuilder software checks the call against customer-defined criteria in the OFFCCODE table. If the call meets the trigger criteria, the switch performs one of the following:

- queries the SCP for instructions
- ignores the criteria and continues processing the call
- blocks the call and applies AIN final treatment (AINF)
- exits the ***Info_Analyzed*** TDP and prevents any further NetworkBuilder interaction for the current call
- allows the call to continue but prevents NetworkBuilder triggering for the remainder of the call

Supported PICs

Although NetworkBuilder has many PICs, the *Office_Code* trigger is evaluated for LNP purposes through PIC 4 (**Analyze_Information**) only. Table 1-1 shows the PICs, TDPs, EDPs, triggers, and events supplied by NetworkBuilder software for the originating call model.

Table 1-1
Supported PICs, TDPs, EDPs, triggers, and events

PIC	TDP/EDP	Trigger/Event
PIC 1: **O_Null**	***Origination_Attempt*** TDP	*Off_Hook_Immediate* trigger
PIC 3: **Collect_Information**	***O_Feature_Requested*** TDP	*O_Feature_Requested* trigger
	Info_Collected TDP	*Tollfree_Service* (note)
		Offhook_Delay trigger
		Shared_Interoffice_Trunk trigger
		PRI_B-Channel trigger
	O_Abandon EDP	*O_Abandon* event

Note: NetworkBuilder supports Bellcore's *TR-NWT-000533* toll-free service specifications. The ability to subscribe to multiple CAIN groups allows for a service integration of the CAIN triggers and the IN/1 *Tollfree_Service* trigger defined in *TR-NWT-000533*.

—continued—

Table 1-1
Supported PICs, TDPs, EDPs, triggers, and events (continued)

PIC	TDP/EDP	Trigger/Event
PIC 4: **Analyze_Information**	***Info_Analyzed*** TDP	*Specific_Feature_Code* trigger
		Customized_Dialing_Plan trigger
		Specific_Digit_String trigger
		Office_Code trigger
	O_Abandon EDP	*O_Abandon* event
PIC 5: **Select_Route**	***Network_Busy*** TDP	*Network_Busy* trigger
	Network_Busy EDP	*Network_Busy* event
	O_Abandon EDP	*O_Abandon* event
PIC 7: **Send_Call**	***O_Term_Seized*** EDP	*O_Term_Seized* event
	O_Called_Party_Busy TDP	*O_Called_Party_Busy* trigger
	O_Called_Party_Busy EDP	*O_Called_Party_Busy* event
	O_Mid_Call TDP	*O_IEC_Reorigination* trigger
	O_Mid_Call EDP	*Switch_Hook_Flash* event
	O_Abandon EDP	*O_Abandon* event
PIC 8: **O_Alerting**	***O_Answer*** EDP	*O_Answer* event
	O_No_Answer TDP	*O_No_Answer* trigger
	O_No_Answer EDP	*O_No_Answer* event
	O_Mid_Call TDP	*O_IEC_Reorigination* trigger
	O_Mid_Call EDP	*Switch_Hook_Flash* event
	O_Abandon EDP	*O_Abandon* event
PIC 9: **O_Active**	***O_Disconnect*** EDP	*O_Disconnect* event

Note: NetworkBuilder supports Bellcore's *TR-NWT-000533* toll-free service specifications. The ability to subscribe to multiple CAIN groups allows for a service integration of the CAIN triggers and the IN/1 *Tollfree_Service* trigger defined in *TR-NWT-000533*.

—continued—

Table 1-1
Supported PICs, TDPs, EDPs, triggers, and events (continued)

PIC	TDP/EDP	Trigger/Event
	O_Mid_Call TDP	*O_IEC_Reorigination* trigger
	O_Mid_Call EDP	*Timeout* event
		Switch_Hook_Flash event
PIC 10: **O_Suspended**	***O_Disconnect*** EDP	*O_Disconnect* event
	O_Mid_Call TDP	*O_IEC_Reorigination* trigger
	O_Mid_Call EDP	*Timeout* event
PIC 11: **T_Null**	***Termination_Attempt*** TDP	*Termination_Attempt* trigger
Note: NetworkBuilder supports Bellcore's *TR-NWT-000533* toll-free service specifications. The ability to subscribe to multiple CAIN groups allows for a service integration of the CAIN triggers and the IN/1 *Tollfree_Service* trigger defined in *TR-NWT-000533*.		
—end—		

SCP interaction

NetworkBuilder call processing interacts with the SCP to determine how the call should be handled. The SCP directs the switch to route the call to the terminating location of the ported number or states that an LNP check was performed but the number was not ported.

Software optionality control

Software optionality control (SOC), part of the DMS Evolution product delivery process, controls the delivery of product computing module loads (PCLs). All features in a PCL are categorized as either base or optional. Base applications are available for immediate use; optional applications are grouped into commercial units called SOC options. SOC options can be purchased by operating companies. LNP uses the CAIN0100 SOC option, as well as the LNP-specific CAIN0700 SOC option, which enables the *Office_Code* trigger to provide LNP services and incoming Home LRN processing. For full functionality of LNP with CAIN queries, CAIN0200 SOC option is also required. For more information on the other NetworkBuilder SOC options, refer to the *UCS DMS-250 NetworkBuilder Application Guide*.

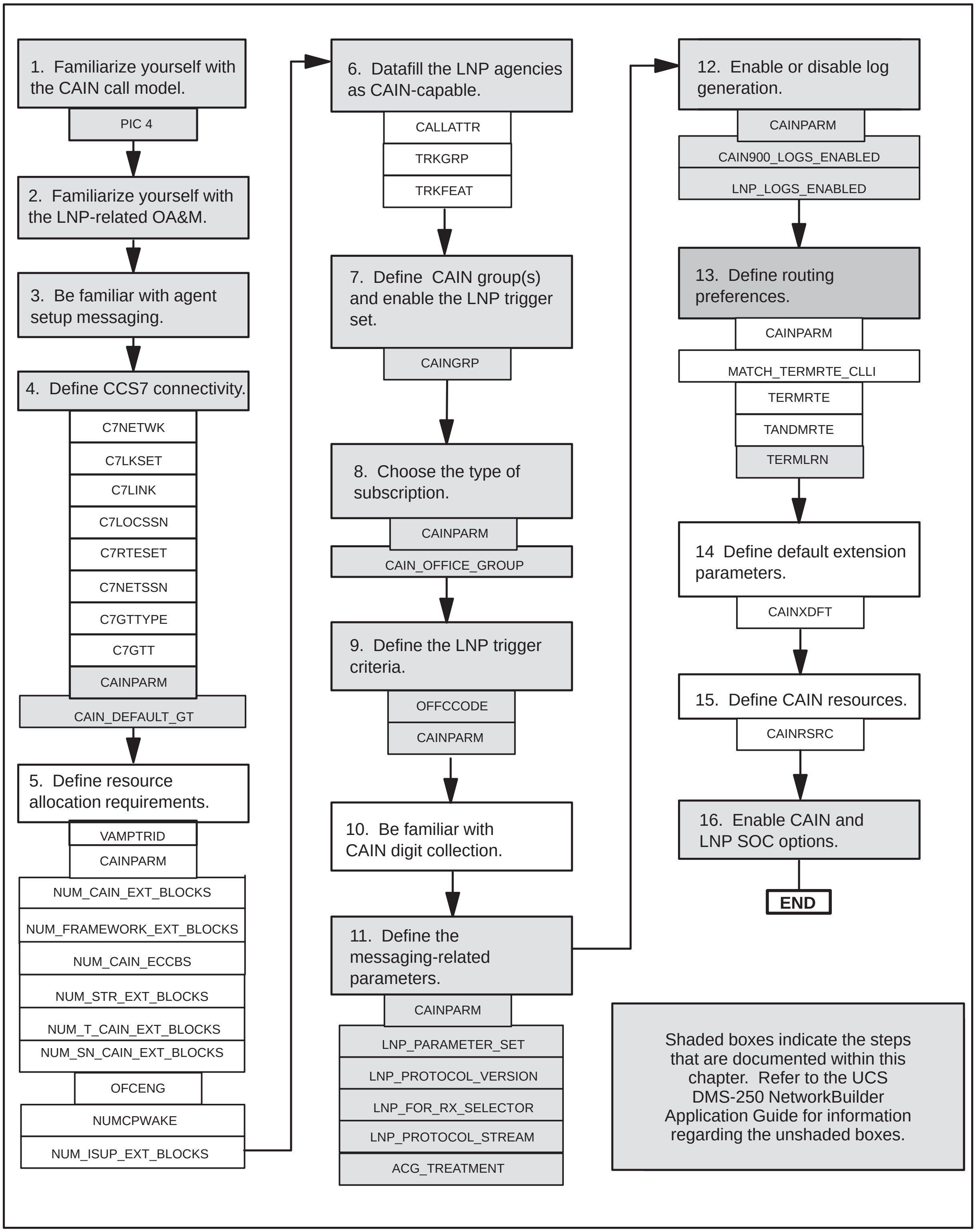
Provisioning NetworkBuilder for LNP

Implementing Local Number Portability (LNP) services requires a base knowledge of table control, the originating call model, Common Channel Signaling 7 (CCS7) network connections, subscription, triggering, and messaging.

This chapter provides the base-level knowledge required to activate LNP services into your network.

Figure 2-1 illustrates the process required to provision Carrier Advanced Intelligent Network (CAIN) for LNP.

Figure 2-1
Provisioning CAIN for LNP



Provisioning

The remainder of this chapter shows you how to provision CAIN for LNP services. Figure 2-2 shows how sections are titled and divided.

Figure 2-2
Example of a provisioning section

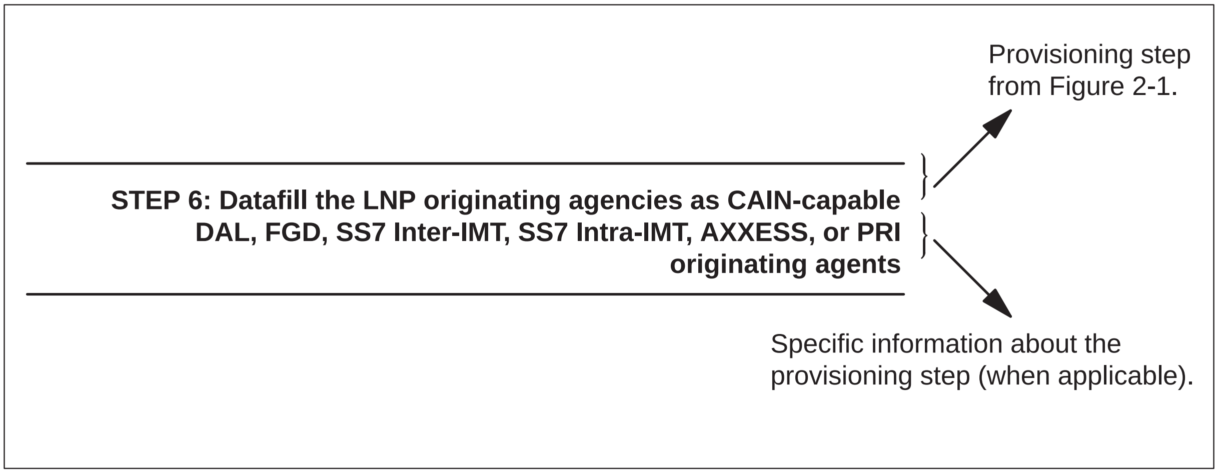


Table 2-1 shows the chapter layout.

Table 2-1
Chapter layout

Step	Title	Page
1	Familiarize yourself with the call model	2-5
	PIC 4: Analyze_Information	2-11
2	Familiarize yourself with LNP-related OA&M	2-23
	Logs	2-24
	Operational measurements	2-28
	Data schema	2-33
	Treatments	2-41
3	Be familiar with agent setup messaging	2-52

Note: Where “no change” is indicated in the Page column, refer to *UCS DMS-250 NetworkBuilder Application Guide* for more information.

—continued—

Table 2-1
Chapter layout (continued)

Step	Title	Page
4	Define CCS7 connectivity	2-53
	CAIN_DEFAULT_GT	2-55
5	Define resource allocation requirements	no change
6	Datafill the LNP agents as CAIN-capable	2-56
	DAL, FGB, FGD, SS7 Inter-IMT, SS7 Inter-IMT in RLT calls, SS7 Intra-IMT AXXESS, or PRI originating agents	2-56
7	Define the LNP CAIN group(s) and enable LNP trigger set	2-58
8	Choose the type of subscription	2-60
9	Define the LNP trigger criteria	2-64
10	Be familiar with CAIN digit collection	no change
11	Define the messaging-related parameters	2-67
	LNP_PARAMETER_SET	2-68
	LNP_PROTOCOL_VERSION	2-69
	LNP_PROTOCOL_STREAM	2-70
	LNP_FOR_RX_SELECTOR	2-71
	ACG_TREATMENT	2-72
12	Enable or disable log generation	2-74
	CAIN900_LOGS_ENABLED	2-74
	LNP_LOGS_ENABLED	2-75
13	Define routing preferences	no change
14	Define default extension parameters	no change
15	Define CAIN resources	no change
16	Enable CAIN and LNP SOC options	2-76

Note: Where "no change" is indicated in the Page column, refer to *UCS DMS-250 NetworkBuilder Application Guide* for more information.

—end—

Step 1: Familiarize yourself with the call model (continued)

Terminology

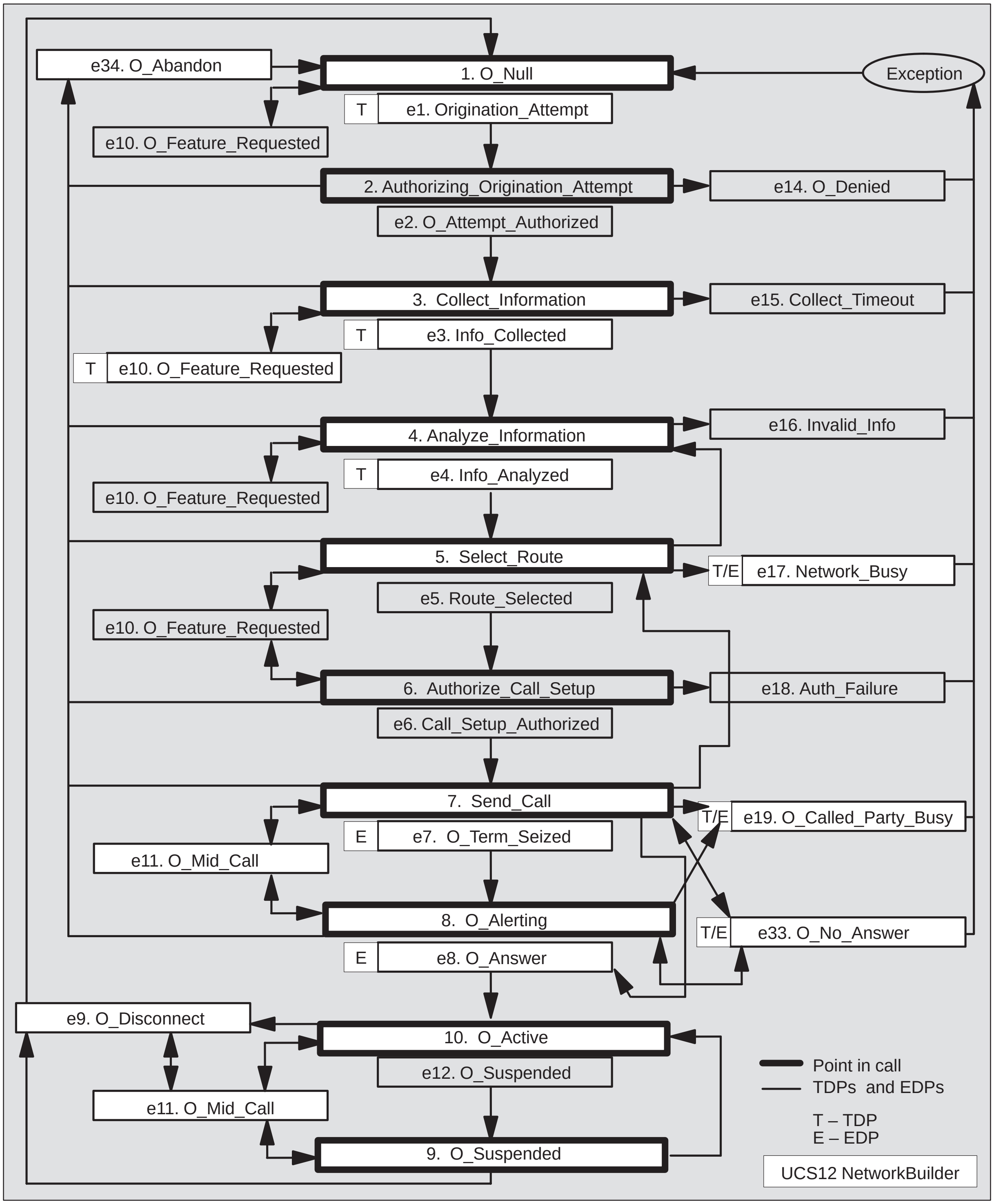
The following terms are related to the call model:

- originating call model – generic representation of a basic call in terms of the processing activities required to establish, maintain, and clear a call.
- EDP – event detection point. EDPs are similar to TDPs. When a call queries the SCP, the SCP may return a conversational message identifying one or more EDPs that occur later in the call model. At this point, the EDP is said to be armed. By arming an EDP, the SCP deactivates triggering until the armed EDP is encountered or deactivated.
- originating call model – generic representation of a basic call in terms of the processing activities required to establish, maintain, and clear a call.
- PIC – point in call. Represents the call processing functionality required by a basic two-party call.
- TDP – trigger detection point. Points within the call model where the switch can temporarily suspend call processing and send a message to the service control point (SCP).
- terminating call model – generic representation of a basic call in terms of the processing activities required terminate a call.
- trigger criteria – identifies call conditions that must be met in order for the switch to perform a specified action.

Figure 2-3 shows the originating call model.

Step 1: Familiarize yourself with the call model (continued)

Figure 2-3
NetworkBuilder originating call model



Step 1: Familiarize yourself with the call model (continued)

Table 2-2 defines the originating call model.

Table 2-2
Call model defined

PIC	Detection Points	Definition
1. **O_Null**		Occurs upon call clearing, after exception handling, or system initialization. At this time, the originating agent is idle and no call exists.
	Origination_ Attempt TDP (Note 1)	Encountered when a call is initiated on the originating agent: • PTS agents – trunk is seized • SS7 agents – switch receives an IAM • PRI agents – switch receives a SETUP message
2. **Authorize_Origination _Attempt**		Corresponds to any screening performed based upon originating agent datafill.
3. **Collect_Information**		Corresponds to the switch collecting enough data (called party address, ANI, authcode) to process the call. The switch collects data based on the dialing plan.
	O_Feature_ Requested TDP	Encountered when the address digits are collected.
	Info_Collected TDP	Encountered when the switch completes the dialing plan.
	O_Abandon EDP	Encountered when the calling party disconnects before the called party answers.
4. **Analyze_Information**		Entered when the switch begins analyzing and translating the collected digits to determine a route index.
	Info_Analyzed TDP	Encountered once the switch has attempted to identify a route index.

Note 1: The switch only evaluates the ***Origination_Attempt*** TDP for calls originating from a DAL (loop or ground starts).

—continued—

Step 1: Familiarize yourself with the call model (continued)

Table 2-2
Call model defined (continued)

PIC	Detection Points	Definition
	O_Abandon EDP	Encountered when the calling party disconnects before the called party answers.
5. **Select_Route**		Entered when the switch selects a terminating agent from the route list that was determined in-switch or by the SCP.
	Network_Busy DP	Encountered when the triggering switch has attempted all routes and is unable to terminate the call or when an SS7 REL message or ISDN Release message is received indicating a network busy condition.
	O_Abandon EDP	Encountered when the calling party disconnects before the called party answers.
6. **Authorize_Call_Setup**		Entered as the switch verifies the authority of the calling party to place the call.
7. **Send_Call**		Entered as the switch attempts to terminate the call.
	O_Term_Seized EDP	Encountered when a terminating trunk is seized on the UCS DMS-250 switch.
	O_Called_Party_Busy DP	Encountered when the switch • attempts to route to a busy DAL or AXXESS agent (with ONNETTRK=Y) • receives an indication from the terminating agent that the end-user is busy
	O_Mid_Call TDP	Encountered when the calling party indicates reorigination.

Note 1: The switch only evaluates the ***Origination_Attempt*** TDP for calls originating from a DAL (loop or ground starts).

—continued—

Step 1: Familiarize yourself with the call model (continued)

Table 2-2
Call model defined (continued)

PIC	Detection Points	Definition
	O_Abandon EDP	Encountered when the calling party disconnects before the called party answers.
8. **O_Alerting**		Entered as the terminating switch applies ringing and ring-back.
	O_Answer EDP	Encountered when the terminating party answers the call.
	O_No_Answer DP	Encountered when the called party does not answer within the specified amount of time.
	O_Mid_Call TDP	Encountered when the calling party indicates reorigination.
	O_Mid_Call EDP for the *Switch_Hook_Flash* event	Encountered during a valid call configuration when a valid call leg presses the asterisk "*" key for more than .6 seconds.
	O_Abandon EDP	Encountered when the calling party disconnects before the called party answers.
9. **O_Active**		Entered when answer indication is received from the called party and when the called party is reconnected after the switch receives an SS7 RESUME (RES) message.
	O_Disconnect EDP	Encountered when reorigination is detected or when a party disconnects.
	O_Mid_Call TDP	Encountered when the calling party indicates reorigination.
	O_Mid_Call EDP for the *Timeout* event	Encountered when the call has been active longer than the specified amount of time.

Note 1: The switch only evaluates the ***Origination_Attempt*** TDP for calls originating from a DAL (loop or ground starts).

—continued—

Step 1: Familiarize yourself with the call model (end)

Table 2-2
Call model defined (continued)

PIC	Detection Points	Definition
10. **O_Suspended**	***O_Mid_Call*** EDP for the *Switch_Hook_Flash* event	Encountered during a valid call configuration when a valid call leg presses the asterisk “*” key for more than .6 seconds.
		Entered when the switch receives an SS7 SUSPEND (SUS) message.
	O_Disconnect EDP	Encountered when reorigination is detected or when a party disconnects.
	O_Mid_Call TDP	Encountered when the calling party indicates reorigination.
	O_Mid_Call EDP *Timeout* event	Encountered when the call has been active longer than the specified amount of time.

Note 1: The switch only evaluates the ***Origination_Attempt*** TDP for calls originating from a DAL (loop or ground starts).

—end—

Analyze_Information PIC

Call processing enters **Analyze_Information** once the dialing plan is completed and the switch has performed the following:

- collected data from the initial address message (IAM) of the SS7 originating FGD, IMT, or AXXESS trunks
- collected digits from a PTS originating DAL, FGB, FGD, or AXXESS

  Note: Refer to *UCS DMS-250 CAIN/FlexDial Interactions* for more information on AXXESS agents.

- validated N00 digits
- collected data from the SETUP message of the originating primary rate interface (PRI) agency
- performed intelligent network (IN)/1 authorization code screening

> **ATTENTION**
> NetworkBuilder call processing always encounters ***Info_Analyzed*** whether or not a route index is identified through normal translations (where the called party number is provisioned in a translation table such as HNPACONT). By not identifying a route index, the switch allows the SCP to control routing. When the switch does not identify a route index, the SCP must provide a route index in an **`Analyze_Route`** message. If the SCP responds with a **`Continue`** message, treatment is set to indicate a translation failure.

Info_Analyzed TDP

Call processing encounters ***Info_Analyzed*** when the following actions occur:

- called party address is available
- call type is available
- IN/1 screening of N00, account codes, authorization codes, and speed dial numbers (as required) is performed
- possible route choices are calculated from the collected information

Step 1: Familiarize yourself with the call model
PIC 4: Analyze_Information (continued)

Instead of immediately seizing an outgoing agent, NetworkBuilder directs the switch to check for the following:

- call subscription to a CAIN group through table STDPRTCT, authcode tables, ANI tables, TRKGRP, CALLATTR, or CAINPARM

Note: CAIN group subscription for AXXESS agents is handled differently; for more information, refer to *UCS DMS-250 CAIN/FlexDial Interactions*.

- enabled triggers (Triggers are enabled through table CAINGRP.)

Note: NetworkBuilder software supports the *Specific_Feature_Code*, *Customized_Dialing_Plan*, *Specific_Digit_String*, and *Office_Code* triggers at the ***Info_Analyzed*** TDP.

When the INFOANALYZED_FOR_RLT parameter in table CAINPARM is set to Y, ***Info_Analyzed*** triggers can be evaluated on the second leg of an SS7 Inter-IMT RLT call. See Chapter 3, “Feature Interactions” for more information.

Subscription

The subscription method is determined in the following order:

1. On reorigination only, SCP-returned CAIN groups, enabling the *Specific_Feature_Code* trigger
2. On reorigination only, SCP-returned CAIN groups, enabling the *Customized_Dialing_Plan* trigger
3. On reorigination only, SCP-returned CAIN groups, enabling the *Specific_Digit_String* trigger
4. On reorigination only, SCP-returned CAIN groups, enabling the *Office_Code* trigger
5. Address subscription, enabling the *Specific_Feature_Code* trigger
6. Address subscription, enabling the *Customized_Dialing_Plan* trigger
7. Address subscription, enabling the *Specific_Digit_String* trigger
8. Address subscription, enabling the *Office_Code* trigger
9. Authorization code subscription, enabling the *Specific_Feature_Code* trigger
10. Authorization code subscription, enabling the *Customized_Dialing_Plan* trigger
11. Authorization code subscription, enabling the *Specific_Digit_String* trigger

Step 1: Familiarize yourself with the call model
PIC 4: Analyze_Information (continued)

12 Authorization code subscription, enabling the *Office_Code* trigger

13 ANI subscription, enabling the *Specific_Feature_Code* trigger

14 ANI subscription, enabling the *Customized_Dialing_Plan* trigger

15 ANI subscription, enabling the *Specific_Digit_String* trigger

16 ANI subscription, enabling the *Office_Code* trigger

17 Agent subscription, enabling the *Specific_Feature_Code* trigger

18 Agent subscription, enabling the *Customized_Dialing_Plan* trigger

19 Agent subscription, enabling the *Specific_Digit_String* trigger

20 Agent subscription, enabling the *Office_Code* trigger

21 Office subscription, enabling the *Specific_Feature_Code* trigger

22 Office subscription, enabling the *Customized_Dialing_Plan* trigger

23 Office subscription, enabling the *Specific_Digit_String* trigger

24 Office subscription, enabling the *Office_Code* trigger

Once a subscription method is identified and a *Specific_Feature_Code*, *Customized_Dialing_Plan*, *Specific_Digit_String*, or *Office_Code* trigger is enabled, the NetworkBuilder performs the following steps:

1 Evaluates trigger criteria (datafilled in the trigger table) against gathered data.

2 Performs the datafilled trigger action.

- — If the trigger action is QUERY, an **`Info_Analyzed`** query message is sent to the SCP.
- — If the trigger action is IGNORE, subscription method determination continues at the point where the trigger was evaluated.
- — If the trigger action is BLOCK, subscription method determination stops, the call is blocked by call processing, and AINF treatment is applied.
- — If the trigger action is LEAVE_TDP, call processing exits the ***Info_Analyzed*** TDP with no further evaluation.
- — If the trigger action is CONT_NOTRIG, call processing exits the ***Info_Analyzed*** TDP and prevents any further NetworkBuilder interaction for the current call. This is reset for reorigination.

3 If the SCP responds with a **`Continue`** message, subscription method determination continues at the point where the trigger was evaluated.

Step 1: Familiarize yourself with the call model
PIC 4: Analyze_Information (continued)

Note: LNP does not impact the *Specific_Feature_Code*, *Customized_Dialing_Plan* or *Specific_Digit_String* triggers. For more information on these triggers, refer to the *UCS DMS-250 NetworkBuilder Application Guide*.

Office_Code trigger

ATTENTION
The *Office_Code* trigger requires the CAIN0700 SOC option. Refer to *UCS DMS-250 NetworkBuilder Application Guide* for more information.

Uses

When you provision NetworkBuilder services on the switch, major call processing decisions can be made by the SCP. You define the services offered on the SCP. The service used by the *Office_Code* trigger is LNP.

Supported originating agencies

The *Office_Code* trigger supports the following originating agencies:

- DAL
- FGB
- FGD
- PRI
- SS7 Inter-IMT
- SS7 Intra-IMT in RLT calls
- AXXESS

Note: Refer to *UCS DMS-250 CAIN/FlexDial Interactions* for more information on support for AXXESS agents.

Subscribing to the Office_Code trigger

Subscription to the *Office_Code* trigger is available on the following:

- SCP-returned basis
- address basis (table STDPRTCT, subtable STDPRT)
- authorization code basis (tables AUTHCODU, AUTHCDU2, AUTHCDU3, AUTHCDU4, or AUTHCDU5)

Step 1: Familiarize yourself with the call model
PIC 4: Analyze_Information (continued)

- ANI basis (tables ANISCUSP or ANIVAL and UNIPROF)
- agent basis (tables TRKGRP, CALLATTR, TRKFEAT)
- office basis (table CAINPARM)

Note 1: An SCP-returned CAIN subscription group is received in **`Analyze_Route`** messages. When a reorigination occurs after an **`Analyze_Route`** message is received, the SCP-returned CAIN group is available for any reoriginated call.

Note 2: CAIN group subscription for AXXESS agents is handled differently; refer to *UCS DMS-250 CAIN/FlexDial Interactions*.

Trigger evaluation

The NetworkBuilder checks the *Office_Code* trigger table (OFFCCODE) and evaluates the call's 10-digit translated address [XLAADDR] against the datafilled range associated with the appropriate CAIN group.

Trigger actions

NetworkBuilder call processing supports the following actions for the *Office_Code* trigger:

- BLOCK – Prevents the call from proceeding and applies AINF treatment.
- IGNORE – NetworkBuilder continues checking the remaining subscription methods for *Customized_Dialing_Plan*, *Specific_Digit_String*, and *Office_Code* triggers. If datafill does not enable the trigger, call processing continues through the call model.
- QUERY – The switch builds an **`Info_Analyzed`** query and sends it to the SCP. The SCP analyzes the received call data and assists the switch with call processing.
- LEAVE_TDP – NetworkBuilder call processing exits the ***Info_Analyzed*** TDP with no further evaluation.
- CONT_NOTRIG – NetworkBuilder call processing exits the ***Info_Analyzed*** TDP and prevents any further NetworkBuilder interaction for the call. This prevention is reset upon reorigination.

Error actions

When a fatal application error occurs during an LNP query, the default error action, ROUTE, is performed. The error action is not datafillable in the trigger table.

Step 1: Familiarize yourself with the call model
PIC 4: Analyze_Information (continued)

The following figure shows how a FGD 1+ call progresses through the call model, encounters ***Info_Analyzed***, and queries the SCP.

Figure 2-4
FGD 1+ call

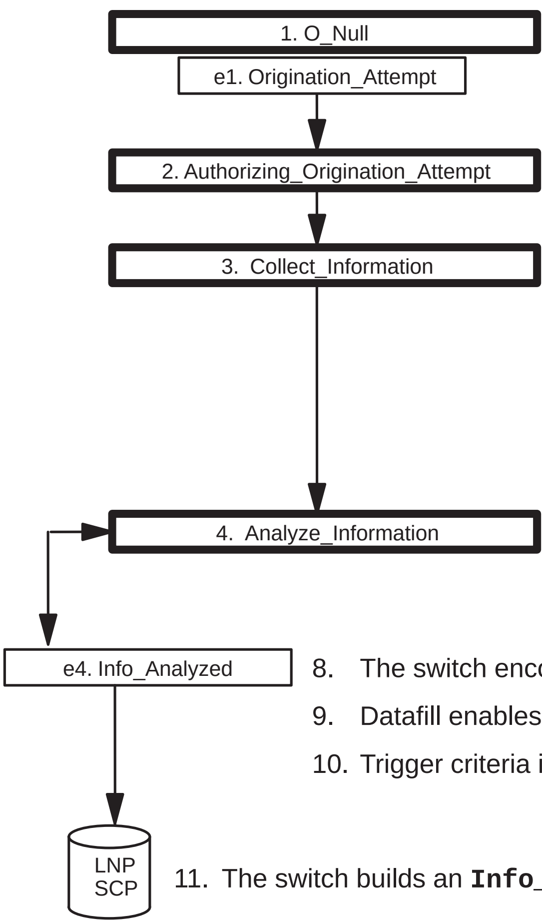
FGD origination Datafill:

Signaling: MF at origination
DTMF for digit collection

Dialing plan:
ADDR: (214)684-1111

Table OFFCCODE

OFFCCODEKY	TODIGS	ACTION		OPTION
CAINGRP1 214	2146841111	QUERY	NPA	$



1. The switch detects seizure and enters **O_Null**.
2. Call processing enters **Authorizing_Origination_Attempt**.
3. Call processing enters **Collect_Information**.
4. The switch collects KP + II + ANI + ST.
5. The switch screens II and ANI digits.
6. The switch collects KP + ADDR + ST.
7. The switch performs digit screening (in-switch or off-board).
8. The switch encounters ***Info_Analyzed***.
9. Datafill enables *Office_Code* trigger.
10. Trigger criteria is met.
11. The switch builds an **`Info_Analyzed`** query and sends it to the LNP SCP.

Step 1: Familiarize yourself with the call model
PIC 4: Analyze_Information (continued)

The following figure shows how an 888 call progresses through the call model, encounters ***Info_Analyzed***, and queries the SCP.

Figure 2-5
888 call

Signaling: MF at origination
DTMF for digit collection

FGD origination
Datafill:

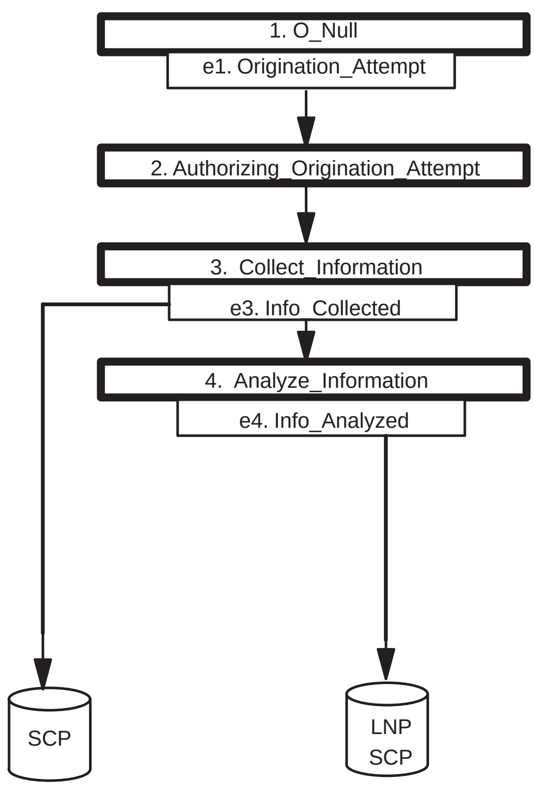
Dialing plan:
ADDR: (888)684-1000
XLAADDR: (214)685-1234

Table OFFCCODE

OFFCCODEKY	TODIGS	ACTION		OPTION
CAINGRP1 214685	214685	QUERY	NPA_NXX	$

Table SIOTRK

SIOTRKKY	TODIGS	ACTION		OPTION
CAINGRP1 888	888	QUERY	SIO_N00	$



1. The switch detects seizure and enters **O_Null**.
2. Call processing enters **Authorizing_Origination_Attempt**.
3. Call processing enters **Collect_Information**.
4. The switch collects KP + II + ANI + ST.
5. The switch screens II and ANI digits.
6. The switch collects KP + ADDR + ST.
7. The switch encounters ***Info_Collected***.
8. Datafill enables *Shared_InterOffice_Trunk* trigger.
9. Trigger criteria is met.
10. The switch builds an ***Info_Collected*** query and sends it to the SCP.
11. **Analyze_Route** with ***`CalledPartyID`*** of (214) 685-1234 was received from the SCP.
12. Call processing encounters ***Info_Analyzed***.
13. Datafill enables *Office_Code* trigger.
14. Trigger criteria is met.
15. Critical call-related information is saved from the initial **`Analyze_Route`** message.
16. Switch builds ***`Info_Analyzed`*** query to LNP SCP.

Step 1: Familiarize yourself with the call model
PIC 4: Analyze_Information (continued)

Info_Analyzed TDP-Request

An **`Info_Analyzed`** TDP-Request is sent to the SCP in a query package with a component type of Invoke_Last. Table 2-3 defines the parameters and usage requirements for the parameters the **`Info_Analyzed`** query may contain. These parameters are a subset of the AIN 0.1 **`Info_Analyzed`** parameters for LNP at *Office_Code*.

Table 2-3
LNP Info_Analyzed query message parameters

Parameter	Usage	Definition
UserID	Required	Contains the network identity of the originating agent
BearerCapability	Required	Contains the bearer capability of the call when the message is built
CalledPartyID	Optional	Contains the translated address
TriggerCriteriaType	Required	Contains the triggering criteria for the call. Possible values are: NPA, NPA_NXX, NPA_NXXX, NPA_NXXXX, NPA_NXXXXX, NPA_NXXXXXX, LNP_OFCD
ChargeNumber	Optional	Contains the billing number that would be used to populate the call detail record at this point in call processing
ChargePartyStationType	Optional	Contains the information digits for the call
CallingPartyID	Optional	Contains one of the following (listed in order of precedence): For FGD, SS7 Inter-IMT, SS7 Global IMT calls: Calling_Party_Address from ISUP message, when available For FGD, DAL, and AXXESS calls: valid ANI (information digits are not passed in this parameter) ***Note:*** Refer to *UCS DMS-250 CAIN/FlexDial Interactions* for more information on AXXESS agents. For PRI calls: CLID Valid SNPA value from table TRKGRP

Step 1: Familiarize yourself with the call model
PIC 4: Analyze_Information (continued)

Table 2-3
LNP Info_Analyzed query message parameters (continued)

Parameter	Usage	Definition
Carrier	Optional	Contains the dialed carrier identification code (CIC) or the CIC value (with an indication of: Selected CIC presubscribed and not input by calling party) from table TRKGRP
JursidictionInfo	Optional	Contains the originating switch's LRN.
Note: For detailed descriptions of the LNP query parameters, refer to Chapter 6, "Outgoing message parameters." ***Note:*** CAIN query parameters for AXXESS agents are handled differently. Refer to *UCS DMS-250 CAIN/FlexDial Interactions* for more information.		
—end—		

Once the query message is built, NetworkBuilder formats the message and sends it to the message encoder. Once encoded, the message is sent to the LNP SCP.

LNP SCP response processing

The following response messages are supported:

- **Analyze_Route**
- **Send_To_Resource** – This response message is used for special processing; no digit collection is performed.
- **Continue** – NetworkBuilder continues checking the remaining subscription methods for *Specific_Feature_Code*, *Customized_Dialing_Plan*, *Specific_Digit_String*, and *Office_Code*. If datafill does not enable any trigger, call processing continues through the call model.
- **Disconnect**

  Note: The **Analyze_Route** or **Continue** message is not expected to contain a **Request_Report_BCM_Event** non-call-related component. If the message contains this component, the transaction is closed and a **Close** message with a ***CloseCause*** value of unexpectedCommunication is sent.

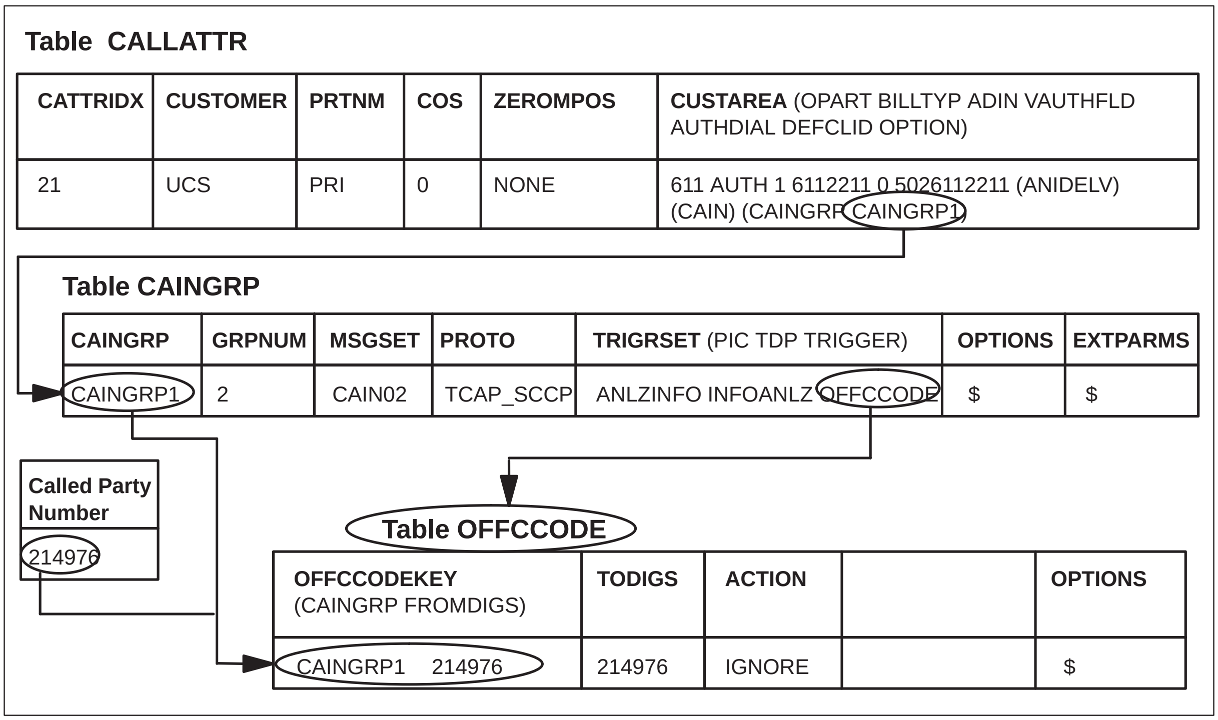
Refer to the *UCS DMS-250 NetworkBuilder Application Guide* for EDP-specific message information.

Step 1: Familiarize yourself with the call model
PIC 4: Analyze_Information (continued)

Datafill

The following figure shows how a subscription table interacts with the *Office_Code* trigger table (OFFCCODE).

Figure 2-6
Subscription-OFFCCODE table interaction



Provisioning the Office_Code trigger

At the CI prompt

1 Provision the originating agency.

2 Subscribe to a CAIN group (table STDPRTCT, authcode tables, ANI tables, TRKGRP, CALLATTR, TRKFEAT, or CAINPARM)

Note: CAIN group subscription is handled differently for AXXESS agents, refer to *UCS DMS-250 CAIN/FlexDial Interactions*.

3 Datafill CAIN group trigger subscription to the (ANLZINFO INFOANLZ OFFCCODE) trigger set (table CAINGRP).

4 Enter table OFFCCODE.

Step 1: Familiarize yourself with the call model PIC 4: Analyze_Information (continued)

5 Define the trigger criteria for a CAIN group by using the following format:

>ADD offccodekey todigs action options
where

offccodekey is comprised of two subfields: CAINGRP and FROMDIGS, where

CAINGRP is the CAIN group requiring Office_Code trigger criteria (from table CAINGRP).

FROMDIGS is the first number used to define the range of the collected address.

Note: The entry in this field should consist of a 3- or 6- to 10-digit number. Entries containing 4 to 5 digits do not correspond to the numbering plan and, therefore, to LNP.

todigs is the second number used to define the range of the collected address.

Note: The entry in this field should consist of a 3- or 6- to 10-digit number. Entries containing 4 to 5 digits do not correspond to the numbering plan and, therefore, to LNP.

action is comprised of 2 subfields: TRIGACT and TRIGCRIT, where

TRIGACT is the trigger action taken when the address is within the FROMDIGS-TODIGS range (BLOCK, IGNORE, QUERY, LEAVE_TDP, CONT_NOTRIG).

If you datafill TRIGACT as:	Then:
BLOCK	Datafill is complete.
IGNORE	Datafill is complete.
QUERY	Go to step 6.
LEAVE_TDP	Datafill is complete.
CONT_NOTRIG	Datafill is complete.

Sample entry: **>ADD caingrp1 222 222 block $**

options No options are available.

Step 1: Familiarize yourself with the call model
PIC 4: Analyze_Information (end)

6 Datafill the TRIGCRIT refinement, where

TRIGCRIT	is only allowed when the ACTION is QUERY. Enter the trigger criteria (NPA, NPA_NXX, NPA_NXXX, NPA_NXXXX, NPA_NXXXXX, NPA_NXXXXXX, LNP_OFCD).

Sample entry: **>ADD caingrp1 222 222 query npa $**

Office_Code trigger criteria is defined.

Step 2: Familiarize yourself with LNP-related OA&M (end)

In order to fully understand, implement, and maintain LNP, you need to be familiar with the logs, operational measurements (OMs), data schema, treatments, billing, and commands used to process LNP calls.

This section provides a brief overview of operating, administrative, and maintenance (OA&M) functions, which include the following:

- Logs – records of call activities that occur within the switch. Refer to *UCS DMS-250 Logs Reference Manual* for more logs information.
- OMs – operational measurements of data that are collected and displayed as the switch performs various operations. Refer to *UCS DMS-250 Operational Measurements Reference Manual* for more OMs information.
- Data schema – tables that direct how the LNP calls are processed. Refer to *UCS DMS-250 Data Schema Reference Manual* for more data schema information.
- Treatment – the method by which a call is disconnected or ended.
- Billing – billing records for calls on the switch. Refer to *UCS DMS-250 Billing Records Application Guide* for more billing information.
- Commands – the commands that activate and deactivate LNP functionality.

Step 2: Familiarize yourself with LNP-related OA&M Logs (continued)

Table 2-4 lists the switch-generated logs associated with NetworkBuilder and, therefore, with LNP.

Table 2-4
NetworkBuilder-related logs

Log	Description
AUD620	EXT Dump. Generated when there is a data dump for a CAIN framework extension block.
AUD621	EXT Dump. Generated when there is a data dump of a CAIN extension block.
AUD665	EXT block audit report. Generated when there is a data dump of a CAIN terminating call model (TCAIN) extension block.
CAIN100	Non-Fatal Application Error. Generated when a nonfatal application error is encountered while transacting with the SCP. Specific error details are given in the log text. Call processing attempts to recover and proceeds with normal in-switch routing when nonfatal application errors occur.
CAIN101	Non-Fatal TCAP/SCCP Error. Generated when a nonfatal TCAP/SCCP error is encountered while transacting with the SCP. Specific error details are given in the log text. Call processing attempts to recover and proceeds.
CAIN102	CAIN SOC Access. Generated when the switch tries to access an idle, non-trigger NetworkBuilder SOC.
CAIN200	Fatal Application Error. Generated when a fatal application error is encountered while transacting with the SCP. Specific error details are given in the log text.
CAIN201	Fatal TCAP/SCCP Error. Generated when a fatal TCAP/SCCP error is encountered while transacting with the SCP. Specific error details are given in the log text.
CAIN300	SSP Routing Trouble Report. Generated when it determines the data returned by the SCP is in the correct format but does not correspond with data in the switch database.
CAIN301	ISUP Cause Indicator Received. Generated when the switch receives a REL message (cause 26) indicating that a call has been misrouted to a ported number or a REL message (cause 28) indicating that it received an improperly formatted GAP. Only generated when LNP is active on the call.
CAIN302	SSP Trouble Report. Generated when a call exceeds the number of times it can request information from the SCP.

Note: Refer to the *UCS DMS-250 Logs Reference Manual* for more information.

—continued—

Step 2: Familiarize yourself with LNP-related OA&M Logs (continued)

Table 2-4
NetworkBuilder-related logs (continued)

Log	Description
CAIN900	SCP Simulator Indices. Generated when the SCP simulator sends a response to NetworkBuilder.
CAIN901	SCP Simulator Action. Generated when the SCP simulator receives an error or abort message from NetworkBuilder. No response message from the SCP simulator is required; also produced when an SCP simulator error is detected.
CAIN902	CAIN Subscription Method. Generated to identify the current NetworkBuilder subscription method when evaluating trigger criteria.
CAIN903	CAINGRP To Trigger. Generated to identify the CAIN group associated with the call when the trigger criteria is met.
CAIN904	CAIN Collectible Overridden. Generated when a collectible on the switch is being overridden. The old information is discarded and the new information is retained.
CAIN905	CAIN Requested Event. Generated to identify the PIC, EDP, event, event action and originating trunk group for a call each time a requested event is reached.
CAIN906	OFFCCODE Trigger Blocked by STS. Generated when an *Office_Code* trigger is blocked by the NO_LNP option in table CAINSTS.
CAIN907	Unarmed EDP action. Generated when the switch detects an unarmed event necessary for multi-party call handling while the switch and the SCP are in conversation.
VAMP201	VAMPTRID High Resource Use. Generated when VAMP resources exceed the 75% or 90% usage threshold set in table VAMPTRID. If resource usage exceeds the 75% threshold, but not the 90% threshold, a minor alarm is indicated. If the 90% threshold is exceeded, a major alarm is indicated.
VAMP202	VAMPTRID Resource Overflow. Generated when attempted usage of a VAMP resource exceeds resource allocation set in table VAMPTRID. A critical alarm is indicated.
VAMP203	VAMPTRID Alarm Cleared. Generated after VAMP201 log reports are output indicating high resource usage and once usage of a VAMP resource drops below 75% of the allocation. Indicates the resource usage was lowered because changes occurred in the call volume or call mix, or the resource allocation was increased in table VAMPTRID.
Note: Refer to the *UCS DMS-250 Logs Reference Manual* for more information.	

—continued—

Step 2: Familiarize yourself with LNP-related OA&M Logs (continued)

Table 2-4
NetworkBuilder-related logs (continued)

Log	Description
VAMP301	VAMPTRID ACG Query Blocked. Generated when an outgoing query is blocked due to encountering an ACG control.
VAMP302	VAMPTRID ACG Control List Change. Generated when an ACG control is added, removed, expired, or updated on the ACG control lists.
VAMP303	VAMPTRID ACG Global Reset. Generated when an ACG Global Reset message is received by the switch.
VAMP304	VAMPTRID ACG Infinite Duration. Generated when an ACG control with an infinite duration is present in one of the control lists. Automatically occurs every ten minutes.
VAMP305	VAMPTRID ACG Overflow. Generated when an ACG control list has overflowed.
VAMP306	VAMPTRID ACG Global Outgoing Control. Generated when a Global Outgoing Control is present in the ACG control list. Automatically occurs every five minutes.
VAMP601	VAMPTRID Audit Summary. Generated when errors are found in the audit of a resource pool or free queue. If an audit completes successfully, this log report is not generated.
VAMP602	VAMPTRID Audit Block Error. Generated during a resource pool audit when an individual resource block is found damaged or in an inconsistent state and is recovered and placed in the free queue. Contains the error type that caused the block to be recovered, as well as a dump of the block contents in hexadecimal form.
VAMP603	VAMPTRID Free Queue Rebuilt. Generated when an application's free queue of a resource type must be rebuilt to recover from queue corruption. The rebuilding of a queue temporarily removes the queue from service, which may cause resource seizure attempts to fail.

Note: Refer to the *UCS DMS-250 Logs Reference Manual* for more information.

—continued—

Step 2: Familiarize yourself with LNP-related OA&M Logs (end)

Table 2-4
NetworkBuilder-related logs (continued)

Log	Description
VAMP901	VAMP Inbound Message. Generated when an inbound message is received from VAMP and the VPTRACE CI tool has enabled message monitoring. The outbound messages bounced back by the transport layer are not logged.
VAMP902	VAMP Outbound Message. Generated when an outbound message is sent through VAMP and the VPTRACE CI tool has enabled message monitoring.
Note: Refer to the *UCS DMS-250 Logs Reference Manual* for more information.	
—end—	

Step 2: Familiarize yourself with LNP-related OA&M Operational measurements (continued)

Table 2-5 lists the OMs pegged by the switch in association with LNP.

Table 2-5
NetworkBuilder-related OMs

Group	Register	Description
CAINLNP		Provides reporting for LNP functionality.
	OFCDLOOK	Counts the number of calls subscribed to the *Office_Code* trigger and number of times table OFFCCODE is referenced.
	LNPQUERY	Counts the number of LNP queries sent to the SCP.
	LRNONLNP	Counts the number of times an LNP query is made to the SCP and a response message is received containing the location routing number (LRN).
	INLNPINF	Counts the number of initial address messages received that indicate LNP has been performed by a previous switch.
	LNPDISCD	Counts the number of times LNP has been performed, but the information obtained is discarded because the terminating agent is a non-ISUP agent, or the terminating agent has been datafilled with option SIGPTDNO in table TRKGRP or TRKFEAT.
	DESTFAIL	Counts the number of times the release cause 26, Ported_Dest_Failed, is received by the switch that performed the query. Release cause 26 is received when the terminating switch could not find the subscriber whose address was in the Generic Address Parameter (GAP) received by the switch. Also counts the number of times release cause 28 is received. Release cause 28 is received when the switch receives an improperly formatted GAP.
	BADGAP	Counts the number of times the switch receives a RELease message with a release cause of 28 indicating that an incorrectly encoded GAP for LNP functionality was detected. LNP must be active on the switch.
	BLKBYSTS	Counts the number of times the *Office_Code* trigger is blocked because the NO_LNP option is datafilled against the STS when the ***Info_Analyzed*** TDP is encountered and table CAINSTS referenced.

Note: Refer to *UCS DMS-250 Operational Measurements Reference Manual* for more information.

—continued—

Step 2: Familiarize yourself with LNP-related OA&M Operational measurements (continued)

Table 2-5
NetworkBuilder-related OMs (continued)

Group	Register	Description
	TERMLRN	Counts the number of times the switch processes a Home LRN.
CAINTRIG		Provides a tuple of OMs for each trigger/event. The tuples are: • 0 TOTAL • 1 OFFHKIMM • 3 OFFHKDEL • 4 PRIBCHNL • 5 SIOTRK • 8 SPECFEAT • 10 CUSTDP • 11 SPECDIG • 13 NETBUSY • 14 OCLDBSY • 15 ONOANSW • 16 TERMATT • 20 OFTRREQ • 21 OFFCCODE • 22 OTERMSZE • 23 OANSWRE
Note: Refer to *UCS DMS-250 Operational Measurements Reference Manual* for more information.		
—continued—		

Step 2: Familiarize yourself with LNP-related OA&M Operational measurements (continued)

Table 2-5
NetworkBuilder-related OMs (continued)

Group	Register	Description
CAINTRIG (continued)		• 24 NETBUSYE • 25 OCLDBSYE • 26 ONOANSRE • 27 TIMEOUT • 28 ODISC • 29 OIECREO • 30 OABANDON • 31 SHF
	QUERY	Counts the number of times a call made on a CAIN-capable agency queries the SCP.
	RESPRCVD	Counts the number of responses the switch receives from the SCP.
	BLOCKED	Counts the number of times a call made on a CAIN-capable agency has BLOCK as the trigger action.
	IGNORE	Counts the number of times a call made on a CAIN-capable agency has IGNORE as the trigger/event action.
	NOTRIG	Counts the number of times a call made on a CAIN-capable agency has CONT_NOTRIG as the trigger action.
	LEAVETDP	Counts the number of times a call made on a CAIN-capable agency has LEAVE_TDP as the trigger action.
	ERROR	Counts the number of times a call made on a CAIN-capable agency queries the SCP and receives an error message.
	NO_MATCH	Counts the number of times a call made on a CAIN-capable trunk is evaluated for trigger criteria and datafill does not match.
	QUERYSCU	Counts the number of times a call made on a CAIN-capable trunk has QUERY_SCU datafilled as the trigger action in the trigger table.

Note: Refer to *UCS DMS-250 Operational Measurements Reference Manual* for more information.

—continued—

Step 2: Familiarize yourself with LNP-related OA&M Operational measurements (continued)

Table 2-5
NetworkBuilder-related OMs (continued)

Group	Register	Description
CAINTRIG (continued)	STRCONV	Counts the number of times the switch receives a **Send_To_Resource** message from the SCP.
	RCLRCONV	Counts the number of times the switch sends a **Resource_Clear** message to the SCP.
	EDPSRCVD	Counts the number of times the switch receives a valid **Request_Report_BCM_Event** component.
	EDPREQ	Counts the number of EDP requests sent to the SCP.
	EDPNOTIF	Counts the number of EDP notifications sent to the SCP.
	FEATADDR	Counts the number of times a call made on a CAIN-capable agency has FEAT datafilled as the trigger action in the table OFTRREQ and provisions the ADDR feature processor.
	FEATAUTH	Counts the number of times a call made on a CAIN-capable agency has FEAT datafilled as the trigger action in the table OFTRREQ and provisions the AUTH feature processor.
	FEATCARD	Counts the number of times a call made on a CAIN-capable agency has FEAT datafilled as the trigger action in the table OFTRREQ and provisions the CARD feature processor.
	NEXTRTE	Counts the number of times a call made on a CAIN-capable agency has NEXTRTE as the trigger/event action.
	NXTCNRTE	Counts the number of times a call made on a CAIN-capable agency has NEXTCNRTE as the trigger/event action.
	CTRCONV	Counts the number of conversational **Connect_To_Resource** messages the switch receives from the SCP.
	CCLRCONV	Counts the number of **CTR_Clear** conversational messages sent to the SCP.
	VIPREQ	Counts the number of times the SCP requests VIP handling, for either message type (**Send_To_Resource** or **Connect_To_Resource**) on a per-trigger basis.

Note: Refer to *UCS DMS-250 Operational Measurements Reference Manual* for more information.

—continued—

Step 2: Familiarize yourself with LNP-related OA&M Operational measurements (end)

Table 2-5
NetworkBuilder-related OMs (continued)

Group	Register	Description
	VIPRESP	Counts the number of times a response with a ***ClearCause*** value of normal is sent to the SCP after VIP handling, for either message type (**Resource_Clear** or **CTR_Clear**), on a per-trigger basis.
	CITR	Counts the number of **Call_Info_To_Resource** messages received by the switch, on a per-trigger basis.
	CIFR	Counts the number of **Call_Info_From_Resource** messages sent to the SCP, on a per-trigger basis.
Note: Refer to *UCS DMS-250 Operational Measurements Reference Manual* for more information.		
—end—		

Step 2: Familiarize yourself with LNP-related OA&M Data schema

Table 2-6 lists the data schema tables related to NetworkBuilder. Many of the tables may not directly apply to LNP.

Note: AXXESS agents require different data schema tables. Refer to *UCS DMS-250 CAIN/FlexDial Interactions* for more information.

Table 2-6
NetworkBuilder-related data schema

Table	Description
Agent provisioning	
CALLATTR	CALL ATTRIBUTES. Provision a PRI call attribute as CAIN-capable or T_CAIN-capable
TRKGRP	TRUNK GROUP. Identifies the trunk group as CAIN-capable. Also provides an index into table TRKFEAT for AXXESS agents.
TRKFEAT	TRUNK GROUP FEATURES. Identifies the features available on an AXXESS agent, such as CAIN or TCAIN-capability and related CAIN group.
Announcements and tones	
CAINRSRC	CAIN RESOURCE. Maps the resources identified by the SCP in a **`Send_To_Resource`** or **`Analyze_Route`** (`callBranding` extension parameter) message to an announcement resource available on the switch.
CCS7 connectivity	
C7GTT	CCS7 GLOBAL TITLE TRANSLATION. Maps a translation type (defined in table C7GTTYPE) to a CCS7 network address.
C7GTTYPE	CCS7 GLOBAL TITLE TYPE. Maps a CCS7-defined translation to a network-defined global title translation type.
C7LINK	CCS7 LINK. Makes the association between the physical equipment of the link and the logical view of the link as a member of a linkset.
C7LKSET	CCS7 LINK SET. Defines the characteristics of a linkset. A linkset is a set of links used as a group. Each link carries traffic between the origination point code and a destination point code. The table also defines attributes that are common to all links in the link set. The links are defined in table C7LINK.
C7LOCSSN	CCS7 LOCATION SUBSYSTEM NUMBER. Defines the subsystems located on the switch.
Note: Refer to *UCS DMS-250 Data Schema Reference Manual* for more data schema information.	

—continued—

Step 2: Familiarize yourself with LNP-related OA&M Data schema (continued)

Table 2-6
NetworkBuilder-related data schema (continued)

Table	Description
C7NETSSN	CCS7 NETWORK SUBSYSTEM NUMBER. Provides the set of remote point codes (PC) and subsystems, at the remote PCs, where messages are routed by the SCCP. A PC is a node in the CCS7 network that may be an SSP, an STP, or an SCP. SCCP routes messages to subsystems at the PC, including SCCP management (SCMG) or SCCP itself, for further global title translation.
C7NETWK	CCS7 NETWORK. Describes the signaling networks in use in a switching office.
C7RPLSSN	CCS7 REPLICATE SUBSYSTEM. Provides the set of remote subsystem replicate pairs. It has a one part key, the subsystem name. For each subsystem a list of PC pairs at which the replicated subsystems reside must be given.
C7RSSCRN	CCS7 REMOTE SUBSYSTEM CONCERNED NODE. Provides a list of concerned nodes for a remote subsystem point code combination. The table has a two part key. The first part is the PC and the second part is the subsystem name. The PC and subsystem combination must be datafilled in table C7NETSSN.
C7RTESET	CCS7 ROUTE SET. Associates linksets used as possible routes for each signaling point in the network. An office point code identifies a signaling point within any network. Each office point code must have a routeset. The information in this table records which routes and linksets can carry the signaling information to the destination signaling point. This table is also used for alternate routing decisions.
VAMPTRID	VAMP TRANSACTION IDENTIFIERS. Provisions the key resources used in Carrier AIN messaging, including transaction and component identifiers and message buffers.
General	
CAINPARM	CAIN PARAMETERS. Provides default NetworkBuilder data for the office. The following data is provided: • determines the global title to be used for requery in the event that a query is blocked due to an ACG control being encountered (ACG_OVERFLOW_GT) • determines the treatment to be applied to a call that is blocked by an ACG control when the error action provisioned in the applicable trigger table is set to TREAT (ACG_TREATMENT)

Note: Refer to *UCS DMS-250 Data Schema Reference Manual* for more data schema information.

—continued—

Step 2: Familiarize yourself with LNP-related OA&M Data schema (continued)

Table 2-6
NetworkBuilder-related data schema (continued)

Table	Description
CAINPARM (continued)	• indicates the type of global title encoding to be used for the CAIN_ADDR_GT global title type (ADDR_GT_FORMAT) • determines whether a network busy condition sets GNCT or RTTE treatment (ALLOW_RTTE_TRTMT) • maximum number of conversations allowed during a call (CAIN_CONVERSATION_LIMIT) • default global title value (CAIN_DEFAULT_GT) • when unable to connect to SCP, routes to the alternate SCP defined here (CAIN_DEFAULT_OVERFLOW_GT) • office subscription group (CAIN_OFFICE_GROUP) • determines whether certain messages or parameters are allowed to be sent in outgoing CAIN TCAP packages (CAIN_PROTOCOL_STREAM) • determines the encoding format to use for parameter whose encodings have changed between software releases (CAIN_PROTOCOL_VERSION) • maximum number of times a subscriber can reset dialing (CAIN_STR_RESET_ALLOWED) • SCP request time-out (CAIN_T1_TIMEOUT) • CAIN900-series log generation (CAIN900_LOGS_ENABLED) • indicates the type of global title encoding to be used for the CAIN_CLID_GT global title type (CLID_GT_FORMAT) • default SNPA (DEFAULT_SNPA) • indicates the type of global title encoding to be used for the CAIN_FEAT_GT global title type (FEAT_GT_FORMAT) • controls whether the *Specific_Feature_Code*, *Customized_Dialing_Plan*, and *Specific_Digit_String* triggers are allowed on the second call leg of an SS7Inter-IMT RLT call (INFOANALYZED_FOR_RLT) • controls how the `OverflowRoutingNo` and ***`CalledPartyID`*** parameters are translated (INTL_XLA_TYPE)

Note: Refer to *UCS DMS-250 Data Schema Reference Manual* for more data schema information.

—continued—

Step 2: Familiarize yourself with LNP-related OA&M Data schema (continued)

Table 2-6
NetworkBuilder-related data schema (continued)

Table	Description
CAINPARM (continued)	• indicates whether a new called number should be evaluated against the trigger criteria from table OFFCCODE (LNP_FOR_RX_SELECTOR) • CAIN301 log generation (LNP_LOGS_ENABLED) • indicates whether to use a limited set of parameters in the **`Info_Analyzed`** message for LNP queries (LNP_PARAMETER_SET) • determines whether certain LNP parameters are allowed to be sent in outgoing packages (LNP_PROTOCOL_STREAM) • determines the encoding format to use for LNP parameters (LNP_PROTOCOL_VERSION) • determines whether LNP functionality is enabled on SS7 IMT RLT originating calls (LNP_4IMT_RLT) • CLLI routing requirements (MATCH_TERMRTE_CLLI) • maximum number of times the switch can send a **`Failure_Outcome`** message per call (MAX_FAILURE_OUTCOMES) • maximum number of times a call can send a TDP-Request and/or EDP-Request to the SCP (MAX_NUM_SERIAL_TRIGGERS) • number of extended call condense blocks available (NUM_CAIN_ECCBS) • CAIN extension blocks available (NUM_CAIN_EXT_BLOCKS) • Framework extension blocks available (NUM_FRAMEWORK_EXT_BLOCKS) • number of send notification extension blocks allowed for the given UCS DMS-250 (NUM_SEND_NOTIFICATION_EXT_BLOCKS) • number of Send_To_Resource extension blocks available (NUM_STR_EXT_BLOCKS) • number of T_CAIN extension blocks available (NUM_T_CAIN_EXT_BLOCKS) • O_No_Answer timer value (O_NO_ANSWER_TIMER) • indicates the type of global title encoding to be used for the CAIN_OFCD_GT global title type (OFCD_GT_FORMAT)

Note: Refer to *UCS DMS-250 Data Schema Reference Manual* for more data schema information.

—continued—

Step 2: Familiarize yourself with LNP-related OA&M Data schema (continued)

Table 2-6
NetworkBuilder-related data schema (continued)

Table	Description
CAINPARM (Continued)	• indicates method to store the digits collected by the *O_Feature_Requested* trigger and FLEXDIAL for later retrieval by both NetworkBuilder and FLEXDIAL (OFTRREQ_FLEXTYPE_MAP) • send CIC value in table TRKGRP to the SCP (SEND_CARRIER_FROM_TRKGRP) • connection type to an IP (STR_CONNECTION_TYPE) • determines the maximum time duration in which the IP must respond to an ISDN FACILITY message with the `cancelIPResource` (TDISC_TIMER) • determines the maximum time duration of the STR-Connection (TSTRC_TIMER) • determines TRTMTCD and COMPCODE fields in the CDR to be zapped to Zero for DISCONNECT message (TRTMTCD_COMPCODE_ZAPPED_ZERO)
CAINXDFT	CAIN EXTENSION PARAMETER DEFAULTS. Provides default values for eleven extension parameters. When a relevant extension parameter is missing in an SCP response, NetworkBuilder call processing uses the values defined in table CAINXDFT. The default values for the extension parameters are associated with a call's CAIN group.
CAINSTS	CAIN SERVING TRANSLATION SCHEME. Provides a list of STS values that when matched, prevent the OFFCCODE trigger table lookup from being performed.
CAINPRT	CAIN PRETRANSLATOR. Defines CAINPRT collectibles used by the O_Feature_Requested handler.
CNPREXLA	CAIN PRETRANSLATOR NAME. Maps extension parameter values for the pretranslator which can be returned by the SCP to a valid pretranslator name in table STDPRTCT.
CNPRTNUM	CAIN PRETRANSLATOR NUMBER. Assigns a number to each CAINPRT collectible in table CAINPRT.
OFCENG	OFFICE ENGINEERING. Provides office engineering parameters. The following parameters are used by NetworkBuilder: • determines the number of call processing timers allocated for use by the switch (NUMCPWAKE) • stores information received in an incoming IAM message for later use in a call (NUM_ISUP_EXT_BLOCKS)

Note: Refer to *UCS DMS-250 Data Schema Reference Manual* for more data schema information.

—continued—

Step 2: Familiarize yourself with LNP-related OA&M Data schema (continued)

Table 2-6
NetworkBuilder-related data schema (continued)

Table	Description
Routing	
TANDMRTE	TANDEM ROUTING. Provisions routing through tandem switches within the IEC network to reach the required terminating switch.
TERMRTE	TERMINATION ROUTING. Provision to route to a terminating switch directly connected to the current switch.
TERMLRN	TERMINATION LOCATION ROUTING NUMBER. Provision to assign Home LRNs to a switch. When the switch determines an LRN is a Home LRN assigned to itself, the switch routes the call based on the DN.
SCP simulator	
CAINCONV	CAIN CONVERSATION. Controls SCP simulator interaction during TCAP conversation with the switch.
CAINKEY	CAIN KEY. Determines a range of possible responses for a given three-part key. The range of possible responses is represented by an option vector of indexes into table CAINMTCH.
CAINMTCH	CAIN MATCHING. Determines possible responses from the SCP simulator
CAINRESP	CAIN RESPONSE. Contains response data to return to the switch. The simulator's encoder takes this data and builds a Transaction Capabilities Application Part (TCAP) message.
CAINREXT	CAIN RESPONSE EXTENSION PARAMETERS. Contains the extension parameters used to build a response message.
CAINUID	CAIN USER IDENTIFICATION. Provides symbolic names for trunk groups and switch identifiers used in the simulator. It is similar to table CLLI in function. The use of symbolic names rather than numbers provides enhanced clarity when datafilling the simulator tables.
IN1RESP	IN/1 RESPONSE. Provides data needed to build a response for IN/1 queries by using the SCP simulator.
Subscription	
CAINGRP	CAIN GROUP. Defines a CAIN group, including the group number and trigger sets.

Note: Refer to *UCS DMS-250 Data Schema Reference Manual* for more data schema information.

—continued—

Step 2: Familiarize yourself with LNP-related OA&M Data schema (continued)

Table 2-6
NetworkBuilder-related data schema (continued)

Table	Description
STDPRTCT	STANDARD PRETRANSLATOR CONTROL. Table STDPRTCT is the first table indexed for digit pretranslation when the incoming or two-way trunk group associated with the call is assigned a standard pretranslator name. Call processing then indexes the appropriate STDPRT subtable. The CAINGRP option identifies the CAIN group for address-based subscription.
ANISCUSP	AUTOMATIC NUMBER IDENTIFICATION SCREENING CUSTOMER PROFILE. Assigns a CAIN group to a particular ANI. CAIN/FlexDial Interaction does not support subscription through table ANISCUSP.
UNIPROF	UNIVERSAL PROFILES. Assigns a CAIN group to a particular profile.
AUTHCODU	AUTHCODE DATABASE. Assigns a CAIN group to a particular authorization code. CAIN/FlexDial Interaction does not support subscription through table AUTHCODU.
AUTHCDU2	AUTHCODE DATABASE 2. Assigns a CAIN group to a particular authorization code. CAIN/FlexDial Interaction does not support subscription through table AUTHCDU2.
AUTHCDU3	AUTHCODE DATABASE 3. Assigns a CAIN group to a particular authorization code. CAIN/FlexDial Interaction does not support subscription through table AUTHCDU3.
AUTHCDU4	AUTHCODE DATABASE 4. Assigns a CAIN group to a particular authorization code. CAIN/FlexDial Interaction does not support subscription through table AUTHCDU4.
AUTHCDU5	AUTHCODE DATABASE 5. Assigns a CAIN group to a particular authorization code. CAIN/FlexDial Interaction does not support subscription through table AUTHCDU5.
CAINPARM	CAIN PARAMETERS. Parameter CAIN_OFFICE_GROUP assigns a CAIN group to an office.
CALLATTR	CALL ATTRIBUTES. Assigns a CAIN group to a PRI call attribute.
TRKGRP	TRUNK GROUP. Assigns a CAIN group to a particular non-AXXESS, non-PRI agent. CAIN/FlexDial Interaction does not support subscription through table TRKGRP.

Note: Refer to *UCS DMS-250 Data Schema Reference Manual* for more data schema information.

—continued—

Step 2: Familiarize yourself with LNP-related OA&M Data schema (continued)

Table 2-6
NetworkBuilder-related data schema (continued)

Table	Description
TRKFEAT	TRUNK GROUP FEATURES. Assigns a CAIN group to a particular AXXESS feature set.
Trigger	
CUSTDP	CUSTOMIZED DIALING PLAN. Defines trigger criteria for *Customized_Dialing_Plan*.
NETBUSY	NETWORK BUSY. Defines trigger criteria for *Network_Busy*.
OCLDBUSY	ORIGINATING CALLED PARTY BUSY. Defines trigger criteria for *O_Called_Party_Busy*.
OFFCCODE	OFFICE CODE. Defines the trigger criteria for *Office_Code*.
OFFHKDEL	OFFHOOK DELAY. Defines trigger criteria for *Offhook_Delay*.
OFFHKIMM	OFF HOOK IMMEDIATE. Defines trigger criteria for *Off_Hook_Immediate*.
OFTRREQ	ORIGINATING FEATURE REQUESTED. Defines trigger criteria for *O_Feature_Requested*.
OIECREO	ORIGINATING IEC REORIGINATION. Defines trigger criteria for *O_IEC_Reorigination*.
ONOANSWR	ORIGINATING NO ANSWER. Defines trigger criteria for *O_No_Answer*.
PRIBCHNL	PRI B-CHANNEL. Defines trigger criteria for *PRI_B-Channel*.
SIOTRK	SHARED INTEROFFICE TRUNK. Defines trigger criteria for *Shared_Interoffice_Trunk*.
SPECDIG	SPECIFIC DIGIT STRING. Defines trigger criteria for *Specific_Digit_String*.
SPECFEAT	SPECIFIC FEATURE CODE. Defines trigger criteria for *Specific_Feature_Code*.
TERMATT	TERMINATION ATTEMPT. Defines trigger criteria for *Termination_Attempt*.
TOLLFREE	TOLL-FREE. Defines trigger criteria for Bellcore's *TR-NWT-000533* toll-free service specifications.

Note: Refer to *UCS DMS-250 Data Schema Reference Manual* for more data schema information.

—end—

Step 2: Familiarize yourself with LNP-related OA&M Treatments (end)

Table 2-7 lists the treatments related to NetworkBuilder.

Table 2-7
NetworkBuilder-related treatments

Treatment	Description
AIND	AIN DISCONNECT. Applied when the SCP determines a call should be disconnected (for example, **`Disconnect`** or **`Send_To_Resource`** with a ***`DisconnectFlag`*** is received).
AINF	AIN FINAL. Applied when the switch detects fatal application errors or you datafill a trigger action of BLOCK.
MLNP	MISROUTED LOCAL NUMBER PORTABILITY. Applied when the final switch on the network receives an SS7 RELEASE message with a value of 26 from the LEC. This is an indication of a routing failure of the ported number.

Refer to *UCS DMS-250 Data Schema Reference Manual*, tables TMTMAP and CSEMAP for a full list of treatment codes.

Step 2: Familiarize yourself with LNP-related OA&M Billing (continued)

CAUTION
Changes may affect site functionality
Changes to the billing system require updates to engineering parameters that may affect site functionality. Any changes to the billing system may affect downstream processing of billing records. Nortel Networks recommends that only experienced personnel make changes to the billing system.

LNP CDR field descriptions

Table 2-8 lists CDR fields populated by NetworkBuilder call processing. Refer to the *UCS DMS-250 Billing Records Application Guide* for more specific CDR field information.

Table 2-8
CDR fields

Field name	Field description
ACCTCD	ACCOUNT CODE. This field is populated with the account code digits collected for the call.
ACG	AUTOMATIC CODE GAPPING. This field indicates a call has encountered an ACG control.
ALTBILL	ALTERNATE BILLING NUMBER. This contains an alternate billing number (for example, the CallingPartyID or ANI).
ANISP	AUTOMATIC NUMBER IDENTIFICATION SPILL. This contains the ANI for MF originating agencies, Charge Number for SS7 originating agencies, or Calling Line ID for PRI originating agencies.
ANSTYPE	ANSWER TYPE. This field contains the type of answer detected.
BILLNUM	BILLING NUMBER. This field identifies the billing number for a call.
CAINCT	CARRIER AIN CALL TYPE. This field contains the CAIN call type received from the SCP response within the callType extension parameter.

—continued—

Step 2: Familiarize yourself with LNP-related OA&M Billing (continued)

Table 2-8
CDR fields (continued)

Field name	Field description
CALLDUR	CALL DURATION. This field contains the duration of a call, measured in 10ms ticks. Call duration is measured as the time between called party answer and on-hook by either called party or calling party.
CALLEDNO	CALLED NUMBER. This field is populated with the translated called party digits for a call.
CARRSEL	CARRIER SELECTION. This indicates how the carrier selection was derived.
CIC	CARRIER IDENTIFICATION CODE. This field identifies the long distance carrier for the call.
CLGPTYNO	CALLING PARTY NUMBER. This field identifies the calling party number of an originating SS7 call.
CN1REQ	CAIN FIRST REQUEST. This field contains the first NetworkBuilder trigger/event that sends a TDP/EDP request message to an SCP (Service Control Point).
CN2REQ	CAIN SECOND REQUEST. This field contains the next unique NetworkBuilder trigger/event that sends a TDP/EDP request message to an SCP (Service Control Point).
CN3REQ	CAIN THIRD REQUEST. This field contains the last unique NetworkBuilder trigger/event that sends a TDP/EDP request message to an SCP (Service Control Point).
CN1TREQ	CAIN FIRST REQUEST TOTALS. This field contains the number of times the CN1REQ trigger/event sends a TDP/EDP request message to an SCP (Service Control Point) during the life of a call.

—continued—

Step 2: Familiarize yourself with LNP-related OA&M Billing (continued)

Table 2-8
CDR fields (continued)

Field name	Field description
CN2TREQ	CAIN SECOND REQUEST TOTALS. This field contains the number of times the CN2REQ trigger/event sends a TDP/EDP request message to an SCP (Service Control Point) during the life of a call.
CN3TREQ	CAIN THIRD REQUEST TOTALS. This field contains the number of times the CN3REQ trigger/event sends a TDP/EDP request message to an SCP (Service Control Point) during the life of a call.
CNPREDIG	CALLED PARTY PREFIX DIGITS. This field identifies one of the following as the translated called number's prefix digits. 0 = No prefix digits 1 = 0 Prefix 2 = 01 Prefix 3 = 011 Prefix 4 = 1 Prefix 5–7 = Not used
CNTOTREQ	CAIN REQUEST TOTALS. This field contains the total number of times TDP/EDP request messages were sent to an SCP (Service Control Point) during the life of a call.
—continued—	

Step 2: Familiarize yourself with LNP-related OA&M Billing (continued)

Table 2-8
CDR fields (continued)

Field name	Field description
COSINDEX	CLASS OF SERVICE INDEX. Identifies the index into table COSUS that was used to perform class of service screening.
DIALEDNO	DIALED NUMBER. This field contains the dialed number or a hotline number for hotline calls.
INFODIG	INFORMATION DIGITS. This field contains the information digits.
LATA	Local Access and Transport Area. This field contains the LATA when the switch sends the LATA.
LNPCHECK	LOCAL NUMBER PORTABILITY CHECK. This is an integer field which indicates the use of LNP functionality in the call.
MLTCOSID	MULTI-CLASS OF SERVICE IDENTIFIER. This field contains the index, within table MULTICOS, used to perform multiple COS screening.
OPART	ORIGINATION PARTITION. This field contains the subscriber’s region and can be used to determine if the subscriber is in the home region.
ORIGLRN	ORIGINATING LOCATION ROUTING NUMBER. This field contains an originating LRN.
ORIGPVN	ORIGINATING PRIVATE NUMBER This field contains the customer-defined dialing plan number assigned to the station or the user originating the VPN call.
PINDIGS	PERSONAL IDENTIFICATION NUMBER DIGITS. This field contains the subscriber’s PINDIGS.
PORTEDNO	PORTED NUMBER. This contains the called party address from the GAP if an LRN is present in an incoming SS7 IAM, or the CPA from the ***GenericAddressList*** parameter if the SCP returns an LRN, or the CPA from the ***GenericAddressList*** parameter if the LNP SCP returns an LRN.
PRESIND	PRESENTATION RESTRICTION INDICATOR. This field indicates if the calling party number can be presented to the called party on SS7 calls.
—continued—	

Step 2: Familiarize yourself with LNP-related OA&M Billing (continued)

Table 2-8
CDR fields (continued)

Field name	Field description
PRJCODE	PROJECT CODE. This field indicates the project code value in the Carrier AIN TCAP message received from the SCP.
RTEINDEX	ROUTE INDEX. This field indicates the number (index) obtained from the routing table used to route the call.
RTETAB	ROUTE TABLE. This field identifies the table used to route the call.
SCPBILL	SERVICE CONTROL POINT BILLING. This field is used to correlate SCP and switch billing records.
SLPID	This field indicates the ***AMAslipID*** parameter value in the Carrier AIN TCAP message received from the SCP.
TERMPVN	TERMINATING PRIVATE NUMBER. This field contains the customer-defined dialing plan number assigned to the called party.
TERMLRN	TERMINATING LOCATION ROUTING NUMBER. This field contains the 10-digit terminating LRN for the call when it matches one of the LRNs assigned the switch.
TPART	TERMINATION PARTITION. This field contains the termination region.
TRMTCD	TREATMENT CODE. This field identifies the treatment code applied to the call.
—end—	

LNP and SS7 RLT billing interactions

For SS7 RLT calls that require LNP interaction, CDR fields may be populated differently.

Step 2: Familiarize yourself with LNP-related OA&M Billing (continued)

In Third-Party interaction RLT calls, the services platform makes a call to the desired called party, collects the information, and drops out of the call so that the originator and terminator are bridged together at the bridging switch. CDRs are produced by the host switch and the bridging switch, and when the call bridges, one merged CDR is produced.

- CDR fields affected are: LNPCHECK, PORTEDNO, ORIGLRN, and CALLEDNO
- When two CDRs are merged, and data was captured for the second call leg, the LNPCHECK CDR field for the second call leg is written into the merged CDR’s LNPCHECK field. Data from the PORTEDNO CDR field of the second call leg is written into the merged CDR, when captured. Data from the ORIGLRN CDR field of the second call leg is not written to the merged CDR. The merged CDR’s CALLEDNO CDR field is populated with data from the second call leg.
- When triggering does not occur on the first leg of the call, but does occur on the second leg, the NetworkBuilder-specific CDR fields are populated by the parameters of the second leg of the call.
- When the UCS12 CDR template is used and multiple NetworkBuilder triggers and/or events take place, trigger/event actions need to be recorded in the merged CDR for billing as follows:
  - When triggering occurs on the first leg of a call (on the way to the services platform) and triggering does not occur on the second leg of the call (from the services platform to the called party), then the trigger and/or event CDR fields are copied from the first call leg’s CDR to the merged CDR.
  - When triggering occurs on the first and second legs of the call, the DIALEDNO, BILLNUM, and UNIVACC CDR fields are populated from the first leg of the call. The CALLEDNO, TERMPVN, ORIGPVN, CAINCT, PIC, TDP, TRIGGER, and SCP CDR fields are populated from the second leg of the call.

    Note 1: If the first CDR does not contain the BILLNUM field, then the field is taken from the second CDR.

    Note 2: To be populated in the merged CDR, the BILLNUM and UNIVACC fields in the Generic Digits parameter must be sent in the bridging FAR message.

    Note 3: For the TERMPVN and ORIGPVN fields to be populated, the AMADigitsDialedWC parameter must be received from the SCP.

    Note 4: For the CAINCT field to be populated, the callType extension parameter must be received from the SCP.

Step 2: Familiarize yourself with LNP-related OA&M Billing (continued)

Note 5: Generic Digits received in a GAP message can overwrite information collected in the IAM.

Note 6: If the Generic Digits are not received in the IAM or the Generic Digits parameter contains no digits, the existing population rules for the charge number parameter applies.

- — When at least three triggers and/or events occur on the first leg of a call and the *Office_Code* trigger occurs on the second leg of the call, data from the second call leg's trigger/event CDR fields overwrites the data in the third trigger/event CDR field of the merged CDR. The merged CDR's first two trigger/event fields are populated with data from the first call leg's trigger/event fields.
- — When only one or two triggers and/or events occur on the first leg of a call and the *Office_Code* trigger occurs on the second leg of a call, then data from the trigger/event CDR field(s) of the first leg of the call is copied into the first trigger/event field(s) of the merged CDR. The second call leg's trigger CDR field data is written into the first available trigger/event field of the merged CDR.
- — When at least three triggers and/or events occur on the first leg of a call, the *Office_Code* trigger being one of the triggers that occurs, and the *Office_Code* trigger occurs also on the second leg of a call, then the merged CDR is populated with data from the first call leg's trigger/event fields and the trigger/event field containing the *Office_Code* trigger is incremented.

- When an LNP query takes place at a switch that is not the bridging switch, the CDR produced at the querying switch contains important LNP data. The merged CDR produced by the bridging switch does not contain the LNP data. Therefore, it may be necessary to examine both the CDR produced by the querying switch and the merged CDR to obtain full LNP information.
- Data from first call leg's ORIGPVN, TERMPVN, and CAINCT CDR fields are used to populate the merged CDR.

In Redirection RLT calls, the services platform collects billing information and sends the call off to be routed to the called number. Redirected calls involve only one trunk to connect from the remote switch to the services platform. When the call is sent off to be routed, the trunk connecting the services platform and the remote switch is released. One CDR is produced at the bridging switch. The CDR at the bridging switch contains the correct LNP information for the call. If the LNP query takes place at a switch that is not the bridging switch, the CDR produced at the querying switch will contain the pertinent LNP information, which will not be contained in the

Step 2: Familiarize yourself with LNP-related OA&M Billing (end)

merged CDR produced by the bridging switch. It may be necessary to examine the CDR produced by the querying switch in addition to the merged CDR in order to obtain complete LNP information.

Step 2: Familiarize yourself with LNP-related OA&M Commands (continued)

The NetworkBuilder platform offers several tools for troubleshooting and tracking purposes. Refer to *UCS DMS-250 Commands Reference Manual* for more commands information.

SOC

Nortel Networks uses Software Optionality Control (SOC) to define and deliver software in product computing-module loads (PCL). Nortel Networks categorizes all functionality in a PCL as either base or optional. Base functionality is available for immediate use. Optional functionality is grouped into commercial units called SOC options, which can be purchased by operating companies. SOC options correspond to functional groups and functions and are controlled by Nortel Networks-supplied passwords.

SOC is the tool for managing options in a PCL. These options reside in the software. When an operating company purchases an option, SOC allows the company to monitor and control its use. You can order, activate, and use these options without a software reload or restart.

The following commands are accessible through SOC:

- ASSIGN
- DBAUDIT
- DELETE
- REMOVE
- RESET
- RESET_AUDIT
- SELECT
- VALIDATE

Note: Refer to step 16, “Enable NetworkBuilder and LNP SOC options,” for more information. Refer to the *UCS DMS-250 NetworkBuilder Application Guide* for more information on NetworkBuilder SOCs.

TRAVER

The TRAVER (translation verification) tool simulates a call from a user specified originating trunk to a user-specified address. TRAVER examines and displays translation and routing data for a single call leg.

TRAVER performs the following:

- verifies the translation tables

Step 2: Familiarize yourself with LNP-related OA&M Commands (end)

- aids in debugging and analyzing translation and routing datafill.
- helps determine reasons for unexpected results and changes required to achieve the expected results.

TRAVER is capable of displaying the following:

- tables used to translate and route a call
- treatment
- NetworkBuilder subscription method and CAIN group
- tuple (from the appropriate trigger table) where trigger criteria was met
- limited messaging parameters

Note: For AXXESS agents the FLEXSIM tool is used in place of the Traver tool. Refer to *UCS DMS-250 CAIN/FlexDial Interactions* for more information.

Step 3: Be familiar with agent setup messaging (end)

To better understand how LNP services are performed through NetworkBuilder call processing, you must be familiar with the LNP Signaling System 7 (SS7) setup message parameters received from the LNP-supported originating agencies.

SS7 IAM messages

LNP information is received by NetworkBuilder in the following originating and terminating agent IAM ISUP parameters:

- FCI – Forward Call Indicator. Used to identify special call capabilities and features, such as the LNP check.
- CPN – Called Party Number. Contains the number to use in translations and routing; also identifies the Nature of Address (NOA) and Numbering Plan. For LNP calls, this parameter can contain the Location Routing Number (LRN).
- GAP – Generic Address Parameter. Used to relay different types of address digits. Identifies the associated Nature of Address (NOA) and Numbering Plan. Used by LNP to store the Called Party Address.

  Note 1: More than one GAP parameter can be present in an IAM.

  Note 2: When LNP uses the GAP parameter to house the Called Party Address, the Type of Address value is set to Ported Dialed Number (#C0).
- JIP – Jurisdiction Information Parameter. Used to carry geographic and service provider information associated with the calling party. For LNP, this can be interpreted as the originating switch's LRN.

Step 4: Familiarize yourself with CCS7 connectivity

CCS7 Terminology

The following terms are related to CCS7 connectivity:

- OSI-related:
  - — Common Channeling Signaling 7 (CCS7) – A digital, message-based network signaling standard defined by the CCITT which separates call signaling information from voice channels so that interoffice signaling is exchanged over a separate signaling link.
  - — Integrated Services Digital Network (ISDN) – A network that provides end-to-end digital connectivity using CCS7 to support a wide range of voice and data services to the end-user.
  - — ISDN User Part (ISUP) – The signalling portion of SS7 required to provide voice and non-voice service in ISDN networks.
  - — Message Transfer Part (MTP) – Serves as a transport system providing transfer of signaling messages between nodes in the network. Concerned about the availability of routing to a node identified by a point code; point code to point code messaging; link and point code status management; doesn’t know the contents of the message.
  - — Signaling System 7 (SS7) – An ANSI standard protocol used by networks.
  - — Signaling Connection Control Part (SCCP) – Software protocol acting as an interface between OSI layers. Provides additional functions to the MTP level to enhance the routing capabilities. Concerned with end-to-end application messaging; global title addressing; subsystem status management; does not know contents of message. Used for non-trunk related services. Provides the global title translation function.
  - — Transaction Capabilities Application Part (TCAP) – A service that provides a common protocol for remote operations across the CCS7 network. The protocol consists of message formatting, content rules, and exchange procedures. TCAP provides the ability for the service switching point (SSP). TCAP is used by the integrated services digital network (ISDN) layer facility message to transport service information for transaction signaling, not associated with an active call, over primary rate interface links.
- global title – The application’s address (translated to a PC + SSN).
- point code (PC) – Address of a node in the CCS7 network.
- subsystem number (SSN) – Address of a subsystem (application) at a node in the CCS7 network (for example, CAINTEST).

Step 4: Familiarize yourself with CCS7 connectivity (continued)

- signaling link – The communication channel between 2 nodes.
- link set – 2 or more redundant links between 2 nodes.
- route – Signaling path from one node to another (may go through multiple nodes).
- route set – Set of all routes from one node to another.

CCS7 tables

This section describes the tables used to define CCS7 connectivity.

C7GTT

Table C7GTT (Common Channel Signaling #7 Global Title Table) maps translation types to CCS7 network addresses. Each global title corresponds to the address of an application. The CCS7 network address resulting from global title translations can correspond to an error, a point code, a subsystem number, a PC and an SSN, or a PC and a new GT. If SSN only is supplied, the local PC is assumed.

C7GTTYPE

Table C7GTTYPE (CCS7 Global Title Type) lists the GT supported by this node. Maps a CCS7-defined translation to a network-defined global title translation type.

For further details on CCS7 tables, datafill information, and examples for CCS7 connectivity, refer to the *UCS DMS-250 NetworkBuilder Application Guide*.

Step 4: Define CCS7 connectivity
CAIN_DEFAULT_GT (end)

NetworkBuilder supplies the following four global title types:

- CAIN_CLID_GT (default)
- CAIN_ADDR_GT
- CAIN_FEAT_GT
- CAIN_OFCD_GT

LNP queries use the global title type, CAIN_OFCD_GT. However, this global title is not datafillable in any trigger table or parameter. Global title address digits are obtained from the ***CalledPartyID*** parameter and contain the full offnet number.

Step 6: Datafill the LNP agencies as CAIN-capable

Before any processing can take place on LNP calls, the originating agency must be provisioned as CAIN-capable. LNP queries to the LNP SCP support the following originating agencies: DAL, FGB, FGD, SS7 Inter-IMT, SS7 Intra-IMT in RLT calls, AXXESS, and PRI. NetworkBuilder queries support the following originating agencies: DAL, FGD, SS7 Inter-IMT, SS7 Global-IMT, AXXESS, and PRI.

Note: LNP is not supported on Global-IMT originating agents.

The following table lists the triggers in the originating call model and their supported originating agents:

Table 2-9
Originating triggers and supported agents

	DAL	FGB	FGD	PRI	SS7 INTER-IMT	SS7 INTRA-IMT	SS7 GLOBAL-IMT	AXXESS
Off_Hook_Immediate	√ (Note 1)							√ (Note 2)
O_Feature_Requested	√		√	√				√
Offhook_Delay	√		√	√	√		√	√
Shared_Interoffice_Trunk	√		√		√		√	√
PRI_B-Channel				√				
Specific_Feature_Code	√		√	√	√		√	√

Note 1: The *Off_Hook_Immediate* trigger supports DAL loop starts and ground starts. DAL TIE originations are not supported.
Note 2: The SS7 Intra-IMT agent is only valid when it is the originating agent in an RLT call.
Note 3: The originating AXXESS agent must be provisioned with FXS- or FXO-type signaling, similar to legacy two-wire DAL agents. The AXXESS agent can imitate DAL, PTS FGD, or SS7 FGD signaling for LNP queries and NetworkBuilder queries. AXXESS agent signaling is provisioned in table TRKSIG. Table TRKGRP provides the index into table TRKSIG. For more information on the AXXESS agent, refer to the *UCS DMS-250 CAIN/FlexDial Interactions Application Guide*.

—continued—

Step 6: Datafill the LNP agencies as CAIN-capable (end)

Table 2-9
Originating triggers and supported agents (continued)

	DAL	FGB	FGD	PRI	SS7 INTER-IMT	SS7 INTRA-IMT	SS7 GLOBAL-IMT	AXXESS
Customized_Dialing_Plan	√		√	√	√		√	√
Specific_Digit_String	√		√	√	√		√	√
Office_Code	√	√	√	√	√	√		√
Network_Busy	√		√	√	√		√	√
O_Called_Party_Busy	√		√	√	√		√	√
O_No_Answer	√		√	√	√		√	√
O_IEC_Reorigination	√		√	√	√			√

Note 1: The *Off_Hook_Immediate* trigger supports DAL loop starts and ground starts. DAL TIE originations are not supported.
Note 2: The SS7 Intra-IMT agent is only valid when it is the originating agent in an RLT call.
Note 3: The originating AXXESS agent must be provisioned with FXS- or FXO-type signaling, similar to legacy two-wire DAL agents. The AXXESS agent can imitate DAL, PTS FGD, or SS7 FGD signaling for LNP queries and NetworkBuilder queries. AXXESS agent signaling is provisioned in table TRKSIG. Table TRKGRP provides the index into table TRKSIG. For more information on the AXXESS agent, refer to the *UCS DMS-250 CAIN/FlexDial Interactions Application Guide*.

—end—

For information on datafilling originating CAIN-capable agents, refer to the *UCS DMS-250 NetworkBuilder Application Guide*.

Step 7: Define LNP CAIN groups/enable LNP trigger set

CAINGRP

Datafill table CAINGRP (CAIN Group) to define the originating CAIN group(s) available for LNP service subscription. Each originating CAIN group enables one or more triggers and provides an index into the triggering tables.

Defining an originating CAIN group

At the CI prompt

1 Enter table CAINGRP.

Note 1: Table CAINGRP provisions the CAIN group name and identifies the triggers that are evaluated for the associated customer, agency, or office subscription.

Note 2: Table CAINGRP must be datafilled prior to defining NetworkBuilder subscription.

2 Create a CAIN group by using the following format:

>ADD caingrp msgset proto trigrset options extparms$
where

caingrp	is the name of the CAIN group (0–16 alphanumeric characters).
grpnum	is the number associated with the CAIN group (0 to 4095).
msgset	is the message set (CAIN02).
proto	is the message protocol (TCAP_SCCP).
trigrset	is the multiple-entry vector comprised of three subfields: PIC, TDP, and TRIGGER, where: PIC — is the point in call defined for this CAIN group. TDP — is the trigger detection point defined for this CAIN group and PIC. TRIGGER — is the trigger being defined for this CAIN group, PIC, and TDP.

The following trigger sets are supported:

- O_NULL_ORIGATT OFFHKIMM
- COLLINFO OFTRREQ OFTRREQ
- COLLINFO INFOCOLL OFFHKDEL
- COLLINFO INFOCOLL SIOTRK
- COLLINFO INFOLL PRIBCHNL
- ANLZINFO INFOANLZ SPECFEAT
- ANLZINFO INFOANLZ CUSTDP
- ANLZINFO INFOANLZ SPECDIG
- ANLZINFO INFOANLZ OFFCCODE
- SELROUTE NETBUSY NETBUSY

Step 7: Define LNP CAIN groups/enable LNP trigger set (end)

- SEND_CALL OCLDBUSY OCLDBUSY
- O_ALERTG ONOANSWR ONOANSWR
- O_ACTIVE OMIDCALL OIECREO

Note: A CAINGRP may subscribe to more than one trigger.

options is an option vector. OANSTIME is the only supported option.

extparms identifies the extension parameters to be sent in the query message.

Sample entry: **>ADD lnpgrp1 100 cain02 tcap_sccp (anlzinfo infoanlz offccode) $ $ $**

An originating LNP CAIN group is defined.

For further details on defining a CAIN group, refer to the *UCS DMS-250 NetworkBuilder Application Guide*.

Step 8: Choose the type of subscription

Now that you have identified the ***Info_Analyzed*** TDP that provides LNP services through the SCP, you need to determine the calls that require this service.

Any call originating on an LNP-supported, provisioned agency (DAL, FGB, FGD, SS7 Inter-IMT, SS7 Intra-IMT in an RLT call, AXXESS, or PRI) can subscribe to LNP services.

Subscription methods for calls are evaluated at **O_Null** and **Collect_Information** PICs. Up to six subscription methods (in order of precedence) are stored by NetworkBuilder call processing for the remaining PICs, including the **Analyze_Information** PIC used for LNP subscription. The six types of subscription are:

- SCP-returned CAIN group
- address subscription
- authorization code (authcode) subscription
- ANI subscription
- originating agency subscription
- office subscription

LNP calls can use any method to subscribe to LNP services.

Note: Subscription is handled differently for AXXESS agents; refer to *UCS DMS-250 CAIN/FlexDial Interactions* for more information.

General subscription rules

For LNP calls, when the ***Info_Analyzed*** TDP is encountered, call processing checks the CAIN group to determine if the *Office_Code* trigger is enabled.

The following rules apply to NetworkBuilder subscription

- Up to six CAIN groups (one for each subscription type) are stored by NetworkBuilder call processing for use during the call.
- A CAIN group returned by the SCP in the `cainGroup` extension parameter always overwrites the previously stored SCP-returned CAIN group.
- Addresses collected at ***O_Feature_Requested*** are overwritten by subsequent addresses collected. The last address collected is used for subscription.
- ANIs and authcodes returned by the SCP are never used to determine subscription.

Step 8: Choose the type of subscription (continued)

- When a TDP is encountered, call processing checks the CAIN group for subscription to the appropriate trigger set.

  Note: A trigger set consists of a PIC, TDP, and trigger.

  CAIN group subscription is always checked in the following order:

  – SCP-returned CAIN group
  – Address-provisioned CAIN group
  – Authcode-provisioned CAIN group
  – ANI-provisioned CAIN group
  – Originating agent-provisioned CAIN group
  – Default office-provisioned CAIN group

- When the trigger criteria is met and the datafilled trigger action is QUERY, FEAT, NEXTRTE, NEXTCNRTE, BLOCK, LEAVE_TDP, or CONT_NOTRIG, call processing performs the action.
- When trigger criteria is not met or the datafilled trigger action is IGNORE, call processing checks the next subscribed trigger at the current TDP. Once all triggers have been checked, call processing returns to the first trigger and checks the next subscription method.
- Reorigination does not reset the SCP-returned CAIN group.
- Because each call has only one switch-determined ANI, agent, and office, the CAIN groups stored for these subscription methods are never re-evaluated.
- However, because more than one authcode may be collected for a call, call processing re-evaluates authcode subscription during the **Collect_Information** PIC for a new CAIN group.

Step 8: Choose the type of subscription (continued)

The following table lists digit types available for trigger criteria evaluation at the *Office_Code* trigger.

Table 2-10
Trigger criteria evaluation

Trigger	Info digits	ADIN	ANI/CLID	XLAADDR	ADDR	CIC
OFFCCODE				X		

Note 1: Digits are evaluated in the order shown (Info digits, ADIN, ANI/CLID, XLAADDR, ADDR, CIC).

Note 2: XLAADDR is the default digit type for OFFCCODE; it is not datafilled.

—end—

Step 8: Choose the type of subscription (end)

Subscribing to originating NetworkBuilder services (office)

Figure 2-7 shows an example of subscribing to originating CAIN services on an Office basis.

Figure 2-7
Office subscription-CAINGRP interaction

Table CAINPARM

PARMNAME	PARMVAL
CAIN_OFFICE_GROUP	LNPGRP

Table CAINGRP

CAINGRP	GRPNUM	MSGSET	PROTO	TRIGRSET (PIC TDP TRIGGER)	OPTIONS	EXTPARMS
LNPGRP	3	CAIN02	TCAP_SCCP	ANZLINFO INFOANLZ OFFCCODE	$	$

At the CI prompt

1 Enter table CAINPARM.

2 Replace the defined parameter by typing:

>REP CAIN_OFFICE_GROUP cain_group
where

cain_group is a CAIN group defined in table CAINGRP providing subscription services to the office.

Sample entry: **>REP CAIN_OFFICE_GROUP lnpgrp1**

The office has subscribed to originating NetworkBuilder services.

For further details on CAIN group subscription methods, refer to the *UCS DMS-250 NetworkBuilder Application Guide*.

Step 9: Define the LNP trigger criteria

> **ATTENTION**
> The Office_Code trigger requires the CAIN0700 SOC option. Refer to Chapter 8, “NetworkBuilder tools,” for more information.

When NetworkBuilder call processing encounters the ***Info_Analyzed*** TDP and *Office_Code* trigger, the following steps have been completed:

1. Data required to process the call as an LNP NetworkBuilder call has been gathered.
2. Table CAINGRP has subscribed to the *Office_Code* trigger set.

   Note: The trigger set identifies the PIC, TDP, and trigger: ANLZINFO INFOANLZ OFFCCODE.

Trigger evaluations

NetworkBuilder evaluates call data against the triggering criteria defined in the OFFCCODE trigger table (identified by the trigger set in table CAINGRP). The trigger table identifies the following:

- conditions required to meet the trigger criteria
- action to take once the trigger criteria is met

Trigger criteria

The following list contains the types available for trigger criteria to be sent to the LNP SCP for the *Office_Code* trigger.

Note: There is no enforced correlation between the digit range specified in the table key and the trigger criteria selected.

- NPA
- NPA_NXX
- NPA_NXXX
- NPA_NXXXX
- NPA_NXXXXX
- NPA_NXXXXXX
- LNP_OFCD

Step 9: Define the LNP trigger criteria (continued)

Trigger evaluation order

The *Office_Code* trigger is the last trigger to be evaluated at the ***Info_Analyzed*** TDP (after *Specific_Feature_Code*, *Customized_Dialing_Plan* and *Specific_Digit_String*) for each subscription type.

The following table shows the order in which triggers are evaluated when a CAIN group is stored for each subscription method.

Table 2-11
Trigger evaluation order

TDP	Trigger	Subscription method					
		SCP	Addr	Auth	ANI	Agent	Office
Info_Analyzed	SPECFEAT *Specific_Feature_Code*	1 (note)	5	9	13	17	21
	CUSTDP *Customized_Dialing_Plan*	2 (note)	6	10	14	18	22
	SPECDIG *Specific_Digit_String*	3 (note)	7	11	15	19	23
	OFFCCODE *Office_Code*	4 (note)	8	12	16	20	24

Note: An SCP-returned CAIN group, specified before reorigination, is available at this TDP after reorigination.

—end—

Trigger table

Define the trigger criteria by datafilling the trigger table, which instructs the switch how to process the call.

OFFCCODE

Table OFFCCODE (Office_Code) is used when the *Office_Code* trigger is evaluated.

The trigger criteria consists of a defined digit range. NetworkBuilder evaluates the call against the datafilled criteria. When the criteria is met, call processing performs the datafilled action (valid actions are IGNORE, BLOCK, QUERY, LEAVE_TDP, and CONT_NOTRIG).

Step 9: Define the LNP trigger criteria (end)

The default error action for OFFCCODE is ROUTE.

Trigger actions

You can datafill the following trigger actions:

- IGNORE – Call processing continues through the call model.

  You can datafill IGNORE in any trigger table.
- BLOCK – The switch applies AIN final treatment (AINF) to the call.
- QUERY – The switch builds the **`Info_Analyzed`** query message and sends it to the SCP. The SCP analyzes the call and assists the switch with call processing.
- LEAVE_TDP – The switch exits the ***Info_Analyzed*** TDP and prevents any further NetworkBuilder interaction for the current call.
- CONT_NOTRIG – The switch allows the call to continue but prevents NetworkBuilder triggering for the remainder of the call.

Trigger options

The OPTION field of the OFFCCODE table is an options vector. However, no options apply to LNP at this time; the only valid option is NIL.

For further details on defining the trigger criteria, refer to *UCS DMS-250 NetworkBuilder Application Guide*.

Associated logs

CAIN301, CAIN902, CAIN903, CAIN906

Associated OMs

CAINTRIG, CAINLNP

Step 11: Define the messaging-related parameters (end)

Define the following LNP-specific messaging-related parameters:

- LNP_PARAMETER_SET
- LNP_PROTOCOL_VERSION
- LNP_FOR_RX_SELECTOR
- LNP_PROTOCOL_STREAM
- ACG_TREATMENT

Step 11: Define the messaging-related parameters LNP_PARAMETER_SET (end)

The LNP_PARAMETER_SET parameter in table CAINPARM is used to enhance realtime processing of LNP calls by controlling whether a complete set or minimum set of parameters are sent in an *Office_Code* **Info_Analyzed** message.

Define the LNP_PARAMETER_SET parameter

At the CI prompt

1 Enter table CAINPARM.

2 Replace the parameter by typing:

>REP LNP_PARAMETER_SET parmval
where

parmval is the parameter value (COMPLETE_SET or MINIMUM_SET)

Note: When the LNP_PARAMETER_SET parameter is set to COMPLETE_SET, the ***UserID***, ***BearerCapability***, ***CalledPartyID***, ***TriggerCriteriaType***, ***CallingPartyID***, ***Carrier***, ***ChargeNumber***, and ***ChargePartyStationType*** parameters are sent in the **Info_Analyzed** message. When the parameter is set to MINIMUM_SET, the parameters sent in **Info_Analyzed** are ***UserID***, ***BearerCapability***, ***CalledPartyID***, and ***TriggerCriteriaType***.

Sample entry: **>REP LNP_PARAMETER_SET minimum_set**

The LNP parameter set value is provisioned for NetworkBuilder call processing

Step 11: Define the messaging-related parameters LNP_PROTOCOL_VERSION (end)

The LNP_PROTOCOL_VERSION parameter determines how LNP parameters are encoded.

Note: This parameter does not affect the processing of the ***`ForwardCallIndicator`*** or ***`GenericAddressList`*** parameters which are returned in an **`Analyze_Route`** in response to a NetworkBuilder query.

Define the LNP_PROTOCOL_VERSION parameter

At the CI prompt

1 Enter table CAINPARM.

2 Replace the parameter by typing:

>REP LNP_PROTOCOL_VERSION parmval
where

parmval the protocol version (V1)

Sample entry: **>REP LNP_PROTOCOL_VERSION V1**

The LNP_PROTOCOL_VERSION parameter is provisioned for NetworkBuilder call processing.

Step 11: Define the messaging-related parameters LNP_PROTOCOL_STREAM (end)

The LNP_PROTOCOL_STREAM parameter in table CAINPARM controls the set of parameters that are sent from the SCP to the switch for a particular LNP query.

Note: Unless a software update changes the LNP_PROTOCOL_STREAM parameter values in table CAINPARM for software releases later than UCS08, UCS07 and UCS08 are the only applicable values.

When the LNP_PROTOCOL_STREAM parameter is set to UCS07, only the ***`CalledPartyID`*** parameter is returned in response to *Office_Code* queries. When it is set to UCS08, to interpret the ***`CalledPartyID`*** as a location routing number (LRN), the following parameters must be returned in the response:

- the ***`ForwardCallIndicator`*** parameter with bit M set to 1
- a ***`GenericAddressList`*** parameter containing a `PortedDialedNo`
- a ***`CalledPartyID`*** parameter

Note: The default is UCS07.

Define the LNP_PROTOCOL_STREAM parameter

At the CI prompt

1 Enter table CAINPARM.

2 Replace the parameter by typing:

>REP LNP_PROTOCOL_STREAM parmval
where

parmval the protocol stream (UCS07, UCS08)

Sample entry: **>REP LNP_PROTOCOL_STREAM UCS08**

The LNP_PROTOCOL_STREAM parameter is provisioned for NetworkBuilder call processing.

Step 11: Define the messaging-related parameters LNP_FOR_RX_SELECTOR (end)

The LNP_FOR_RX_SELECTOR parameter in table CAINPARM is used to indicate if NetworkBuilder interactions are valid for calls using the RX selector.

When the *Office_Code* trigger is enabled and the RX selector is encountered, the following conditions apply when the LNP_FOR_RX_SELECTOR parameter is set to “Y”:

- if the new/retranslated number is not the same as the called number, then an evaluation of the *Office_Code* trigger takes place with the new/retranslated number.
- if the new/retranslated number is the same as the called number, then the *Office_Code* trigger will have already been evaluated with that number and no reevaluation of the *Office_Code* trigger takes place with that number.

When the LNP_FOR_RX_SELECTOR parameter is set to “N”, the new/retranslated number is not evaluated against the *Office_Code* trigger. However, if the called number is changed by the RX selector, then any previous LNP data is reset.

Define the LNP_FOR_RX_SELECTOR parameter

At the CI prompt

1 Enter table CAINPARM.

2 Replace the parameter by typing:

>REP LNP_FOR_RX_SELECTOR parmval
where

parmval the parameter value (Y, N)

Sample entry: **>REP LNP_FOR_RX_SELECTOR Y**

The LNP_FOR_RX_SELECTOR parameter value is provisioned for NetworkBuilder call processing.

Step 11: Define the messaging-related parameters ACG_TREATMENT (continued)

The ACG_TREATMENT parameter in table CAINPARM is used to define a treatment for a call that is blocked by an automatic code gapping (ACG) control when the error action equals TREAT in the applicable trigger table.

Note: The error action is taken if the ACG_OVERFLOW_GT parameter equals NIL or if the query was blocked at the overflow global title.

Define the ACG_TREATMENT parameter

At the CI prompt

1 Enter table CAINPARM.

Step 11: Define the messaging-related parameters
ACG_TREATMENT (end)

2 Replace the parameter by typing:

>REP ACG_TREATMENT parmval
where

parmval	the parameter value (UNDT, NOSC, PDIL, PSIG, INAC, CNDT, VACT, MSCA, MSLC, NBLH, NBLN, EMR1, EMR2, UNCA, SYFL, CQOV, HNPI, UNDN, BLDN, BUSY, UNOW, TDND, UNIN, SSTO, NCRT, RODR, MANL, ORSS, TESS, OPRT, TRBL, ANCT, PNOH, DNTR, NECG, FECG, ORMC, TOVD, CONF, RRPA, ORAF, TRRF, ORAC, ORMF, SRRR, DISC, UNPR, BLPR, EMR3, EMR4, NOCN, PMPT, SORD, INAU, TINV, CNOT, DCFC, PRSC, GNCT, ATBS, MHLD, DODT, TDBR, RSDT, PTOF, VACS, ANTO, NMZN, FNAL, UMOB, ERDS, STOB, STOC, EMR5, EMR6, INOC, ANIA, CFWV, NACK, CACE, D950, N950, ILRS, NACD, DACD, ADBF, PGTO, AIFL, FDNZ, CCTO, CCNV, CCNA, FDER, NOSR, CGRO, VCCT, LCAB, INCC, CONP, NINT, SCFL, NCIX, NCII, NCTF, NONT, NCUN, ATDT, ANBB, IVCC, SCUN, INPD, NPAR, IDPB, CNAC,CBTN, MTOC, ANFL, CHAN, CHAF, OSVR, N00B, CFOV, ILRR, COSX, CACB, SINT, IWUC, INBT, NC8F, FRDR, C7AP, DTFL, BBFS, NTRS, CREJ, UPAB, SORE, CNAD, VPFX, CCAP, ACPR, CCIR, ADPA, CCDT, UCCN, CBDN, N9DF, N9OB, N9NS, CCCF, SCRJ, ICNF, LECV, LCNV, CGFL, VPFL, PTFL, SCA, NCS0, NCS1, CHNF, BCNI, RING, JACK, ITCF, NVIP, ACRJ, FCNI, PERR, INVM, SONI, CDAS, CDAF, CDDS, CDDF, AARD, DSCN, GFNV, LBSY, TBSY, IIEC, NOBC, NORA, PER1, PER2, PER3, PER4, PER5, CER1, WUCR, MTBL, MWKP, RFCS, RFCD, RFCE, EROR, ERTR, ERTO, TRGB, ESNF, MBIA, Q33A, Q33B, AIND, AINF, INRF, RTTE, ITDN, PRTO, PAGE, CFWD, LDAA, LDAD, ORBT, AVPF, AVP2, BLMO, CCRG, CCRP, CCRM, CCRH, CCRT, MSOA, RTEE, PSNF, IRET, ISCN, IDST, DDPB, DRET, DDSN, DDST, ONPB, ORET, OSCN, ODST, ZMPB, ZMRT, ZPPB, ZPRT, MAUC)

Sample entry: **>REP ACG_TREATMENT AINF**

The ACG_TREATMENT parameter is defined.

Step 11: Enable or disable log generation

The CAIN900_LOGS_ENABLED parameter allows/disallows the generation of the following CAIN900 logs. LNP uses the CAIN906 log which is generated when an *Office_Code* trigger is blocked by the NO_LNP option in table CAINSTS.

Enable the CAIN906 log

At the CI prompt

1 Enter table CAINPARM.

2 Replace the parameter by typing:

>REP CAIN900_LOGS_ENABLED parmval
where

parmval the enabled log numbers (up to eight values: 900, 901, 902, 903, 904, 905, 906, or 907)

Sample entry: **>REP CAIN900_LOGS_ENABLED 906**

CAIN906 log is enabled.

Step 11: Enable or disable log generation (end)

The LNP_LOGS_ENABLED parameter allows/disallows the generation of the CAIN301 log. The UCS DMS-250 switch generates the CAIN301 log when the switch receives a REL message (cause 26) indicating that a call has been misrouted to a ported number. The UCS DMS-250 switch also generates the CAIN301 log when the switch receives a REL message (cause 28) indicating that the GAP received is not formatted properly. The UCS DMS-250 switch only generates if LNP is active on the call.

Enable the CAIN301 log

At the CI prompt

1 Enter table CAINPARM.

2 Replace the parameter by typing:

>REP LNP_LOGS_ENABLED parmval
where

parmval the enabled log number (301)

Sample entry: **>REP LNP_LOGS_ENABLED 301**

CAIN301 log is enabled.

Step 16: Enable NetworkBuilder and LNP SOC options

Software optionality control (SOC) enables software to be defined and delivered in product computing module loads (PCLs). All functionality in a PCL is categorized as either base or optional. Base functionality is available for immediate use. Optional functionality is grouped into commercial units called SOC options, which can be purchased by operating companies. SOC options correspond to functional groups and functions and are controlled by Nortel Networks-supplied passwords.

The order codes listed in Table 2-9 are necessary for full implementation of LNP.

Table 2-12
NetworkBuilder SOC order codes

Order code	SOC name	Available functionality when SOC is on
CAIN0100	Messages	Usage option for all NetworkBuilder messages to the SCP
CAIN0200	Extension Parms	Extension parameter set

—continued—

Step 16: Enable NetworkBuilder and LNP SOC options (continued)

Table 2-12
NetworkBuilder SOC order codes (continued)

Order code	SOC name	Available functionality when SOC is on
CAIN0700	LNP Query on Origination (QOO)	This SOC enables the following: • OFFCCODE trigger table lookup • setting bit M in the outgoing FCI parameter, based on the SCP response from a NetworkBuilder query (The GAP is not sent.) • incoming Home LRN processing This SOC does not interfere with the ability to do the following: • pass-through an FCI and/or GAP parameter • convert GAP to the called party address when terminating to a non-ISUP trunk • handle ISUP release cause 26 according to existing switch logic • place an LRN in the CALLEDNO field of the CDR when incoming LNP information is received
CAIN0900	Auto Code Gapping	Provides the capability to prevent overloading a given SCP and to reduce the impact of mass calling events.

—end—

Enabling the LNP SOC option

At the CI prompt

1 Enter the SOC command set by typing:

>SOC

2 Assign a key code using the following format:

>ASSIGN RTU keycode TO soc_option
where

keycode is the password that will enable the SOC.
soc_option is the SOC option you want to enable.

Step 16: Enable NetworkBuilder and LNP SOC options (end)

3 Activate the NetworkBuilder SOC state option:

>ASSIGN STATE ON TO soc_option
where

soc_option is the SOC option you want to enable.

Sample entry: **>ASSIGN STATE ON TO cain0700**

A SOC option is enabled.

For information regarding NetworkBuilder SOC options, refer to the *UCS DMS-250 NetworkBuilder Application Guide*.

Feature interactions

The manner of how LNP interacts with other features is documented in the following paragraphs.

1129-style IP

1129-Style IP interacts with LNP in the following ways:

- For SS7 originations at the local switch, the local switch does not sent LNP information received in an incoming IAM to IPs . This applies to IPs which use the SS7 protocol for communication with the local switch.
- The ***`DestinationAddress`*** parameter received in a **`Send_To_Resource`** message cannot be a ported number.
- Because the originating call model is not supported on the intermediate and remote switches, LNP checks are not performed on these switches.

Automatic Code Gapping (ACG)

Automatic Code Gapping – ACG prevents the overloading of a given SCP and reduces the impact of mass calling events. ACG is supported for LNP except for the following:

- Th ACG_Requery extension parameter is not supported for LNP.
- The SSP is unable to send the ACG_Overflow message for LNP.

Note: For more information on ACG, see the *UCS DMS-250 NetworkBuilder Application Guide*.

Call processing

When an LRN is outpulsed as the called party address, all 10-digits of the LRN are outpulsed unless digits are removed during translations by certain selectors in table HNPACONT.

LNP overrides general call processing and the UCS DMS-250 switch does outpulse the NPA for the following reasons:

- The CONNGNPA field in table TRKGRP for the terminator equals the NPA of the ANI (in the case of EANT).

- The SNPA field in table TRKGRP for the originator equals the NPA if the terminator is non-EANT.

DESTFAIL OM

When the switch receiving the LRN cannot locate the subscriber, it returns a release cause 26 (misrouted LNP call). This means that the LRN routing attempt failed. When the GAP for LNP is not formatted correctly, the receiving switch sends a release cause 28 (improperly formatted GAP) . The DESTFAIL OM is pegged in the switch processing the release message. Processing of the release message is then returned to call processing, where the treatment is mapped in table CSEMAP. If no treatment is datafilled for the release value in CSEMAP, then vacant_code treatment is set, and normal treatment processing is continued.

Dialed Number Inward Service (DNIS)

It is not foreseen that GAPs for Dialed Number Inward Service (DNIS) and LNP would be used from a service perspective. The only point of interaction between the two services involves termination to a non-SS7 agent. The LNP philosophy is to put the called party address from the GAP for the LNP into the CalledPartyNumber field of the the outgoing IAM, thus replacing the LRN. The DNIS functionality may override this by placing the DNIS into the CalledPartyNumber field.

Intelligent Network (IN)/1 queries

You should not try to perform an IN/1 query using the LRN.

Mechanized Calling Card Services (MCCS)

MCCS – LNP triggering is supported for MCCS calls in the following manner:

- The switch receives an MCCS call.
- The switch collects the calling card number and validates it either in-switch or by an external database.
- If the calling card number is valid, then a confirmation tone/announcement is played.
- Digits are collected.
- An LNP trigger table (OFFCCODE) check is performed at table OFFCCODE.
- If the trigger criteria is met, an LNP query could be made based on the translated address.
- If the number is ported, then the LNP SCP responds with the LRN.

- If the number is not ported, then the LNP SCP responds with the original translated number.

Non-standard routing

Non-standard routing through table HNPACONT could send a ported number through the network. This number would go to the donor switch where it would fail to route because it was unknown. Use of non-standard routing could impact the LRN being sent in the outgoing message.

Programmable Service Node (PSN)

LNP information is discarded if programmable service node (PSN) is invoked.

Reevaluation

For both direct termination and standard routing, reevaluation at the *Office_Code* trigger will not occur if an LRN is returned or if an FCI parameter with bit M set is returned.

Release Link Trunk (RLT)

RLT – Third-party and Redirect RLT calls are supported. Operator-initiated RLT calls are not supported. For more information on SS7 RLT calls, refer to the *UCS DMS-250 SS7 RLT Application Guide* and *UCS DMS-250 NetworkBuilder Application Guide*.

When the service platform returns a redirecting facility request (FAR) message or IAM with the supplementary line information (SLI) parameter containing the value of `RLTCallOperation`, the message contains the address of the terminating route. A query to the LNP SCP at the *Office_Code* trigger needs to be allowed in order to obtain the LRN to correctly route the call.

NetworkBuilder triggering is not normally allowed for RLT calls. However, for the *Office_Code* trigger to be encountered on an RLT call, the LNPRLT option must be datafilled against the SS7 Inter-IMT or SS7 Intra-IMT agent in table TRKGRP. The LNPRLT option is similar to the CAIN option making the the agent NetworkBuilder-capable, however, the LNPRLT option allows the *Office_Code* trigger to be evaluated for RLT calls only and does not apply to any other NetworkBuilder triggers. Additionally, the CAIN option datafilled against the agent in table TRKGRP allows the *Office_Code* trigger to be evaluated for RLT calls on SS7 Inter-IMTs and SS7 Intra-IMTs agents. It is necessary that the LNPRLT option be available on SS7 Inter-IMT agents to specify that only the *Office_Code* trigger be evaluated in RLT call scenarios where the agent is used as a tandem trunk.

The CAINGRP option is allowed for SS7 Intra-IMT agents to provide agent subscription to a CAIN group enabling the *Office_Code* trigger.

Note: The RLT option in table TRKGRP does not have to be datafilled against the agent in order for the LNPRLT option to be datafilled.

RLT third-party scenarios

For RLT third-party scenarios, when the remote switch receives an IAM containing the SLI parameter with the value `RLTCallOperation` and the *Office_Code* trigger has been enabled (with the CAIN option or LNPRLT option), the Called Party Number parameter is evaluated in table OFFCCODE to check if triggering should be performed. If the criteria is met, the datafilled trigger action is performed. If the action is QUERY, an LNP query (**`Info_Analyzed`**) is launched to the LNP SCP. When the trunk is being used for a Second Call Leg in this manner, only the ***Info_Analyzed*** TDP may be encountered and only the *Office_Code* trigger is evaluated.

RLT redirect scenarios

For RLT Redirect scenarios, when the bridging switch receives a redirecting FAR message and the *Office_Code* trigger has been enabled (with the CAIN option or LNPRLT option on the originating agency), the Called Party Number parameter is evaluated in table OFFCCODE to check if triggering should be performed.

- If triggering is to occur, then an FAA (Facility Accepted) message is sent to the Services Platform unless reorigination resources cannot be allocated, in which case an FRJ (Facility Reject) is always sent.
- If triggering is not to occur and no routes were found with the original called number, then an FRJ message is sent.
- RLT call processing continues with releasing the call. Then, if the call was to trigger, the trigger action datafilled in OFFCCODE is performed. If the action is QUERY, an LNP query (**`Info_Analyzed`**) is launched to the SCP. When the response to the query is received, NetworkBuilder call processing attempts to route the call.
- If triggering was not to occur, then the call proceeds as if there had been no NetworkBuilder interaction. After the call is redirected, the *O_Disconnect* event, and the ***O_No_Answer***, ***Network_Busy***, ***O_Called_Party_Busy***, and ***O_Mid_Call*** DPs are evaluated only if NetworkBuilder is enabled (with the CAIN option, not the LNPRLT option). When encountering any DPs after redirection, only agent and office subscription types are evaluated.
- Reorigination information provided by the Services Platform overrides the reorigination information determined by the ***O_Mid_Call*** DP.

Note: For any RLT calls, the LNP query is not performed when the STS for the call has the NO_LNP option datafilled against it in table CAINSTS.

SS7 Inter-IMT RLT call

For the second leg of an SS7 Inter-IMT RLT call, the second call leg can trigger on *Specific_Feature_Code*, *Customized_Dialing_Plan*, *Specific_Digit_String* , and *Office_Code* at the ***Info_Analyzed*** TDP. The BILLNUM and UNIVACC generic digits parameters can be sent to the SCP from the *Specific_Feature_Code*, *Customized_Dialing_Plan*, and *Specific_Digit_String* triggers.

- If the LNPRLT option is datafilled in table TRKGRP for UCS08 or higher releases and the CAINPARM parameter INFOANALYZED_FOR_RLT is set to Y, then the four triggers at the ***Info_Analyzed*** TDP are allowed.

  Note: *Office_Code* triggering is independent of the INFOANALYZED_FOR_RLT parameter.

- If the LNPRLT option is datafilled in table TRKGRP for UCS08 or higher releases and the CAINPARM parameter INFOANALYZED_FOR_RLT is set to N, then only the *Office_Code* trigger is allowed.

  Note: For more information on NetworkBuilder triggers, refer to the *UCS DMS-250 NetworkBuilder Application Guide*. For information on LNP and SS7 RLT billing interactions, refer to Chapter 2, “Provisioning NetworkBuilder for LNP.”

Reorigination

LNP information is reset upon reorigination. Previous LNP information will not be used by call processing on the reoriginated call.

Table RTEATTR

Routing Based Outgoing Parameter Modifications allows certain parameters to be included or excluded in the outgoing IAM according to the datafill in this table. One of these parameters is the GAP (Generic Address Parameter). If the type of GAP used by LNP is datafilled in table RTEATTR to be excluded (datafilled as GENADDR 192), this can impact the passing of LNP information in the GAP in a tandem scenario. However, if LNP is enabled at the current switch, then LNP overrides the RTEATTR datafill.

LNP_FOR_RX_SELECTOR and LNP_4IMT_RLT

When both the LNP_FOR_RX_SELECTOR and the LNP_4IMT_RLT CAINPARMS are enabled, a double-dip scenario exists where two LNP queries can be initiated for the second leg of an SS7 RLT IMT call if the RX selector is encountered and the called party number is changed.

CIC routing

When routing a call based on incoming CIC information, it is sometimes advantageous to block the LNP lookup for certain calls. In order to block a call from initiating an LNP trigger table lookup, STSs are entered in table CAINSTS.

A call that subscribes to NetworkBuilder services and has the *Office_Code* trigger enabled evaluates table CAINSTS just before evaluating table OFFCCODE. If the call processing software finds the STS for the call in table CAINSTS and the NO_LNP option is datafilled, then the call processing software does not access table OFFCCODE and does not perform a trigger action for the *Office_Code* trigger. The BLKBYSTS OM register is pegged and the CAIN906 log is generated.

If the call processing software finds the the STS for the call in table CAINSTS and the NO_LNP option is not datafilled, the call continues and the call processing software evaluates table OFFCCODE.

SS7 ISUP and SS7 TCAP messaging

Local Number Portability (LNP) and NetworkBuilder use the following Signaling System 7 (SS7) message protocols to provide LNP services:

- ISDN User Part (ISUP)
- Transaction Capabilities Application Part (TCAP)

SS7 ISUP Messages

SS7 messaging is the chosen method of communicating LNP information through the network. LNP uses the following SS7 ISUP parameters to communicate its information.

- Forward Call Indicator (FCI)
- Called Party Number (CPN)
- Generic Address Parameter (GAP)
- Jurisdiction Information Parameter (JIP)

Note: The described use of these parameters is valid from ISUP to ISUP. If some other type of originator or terminator is used, adjustments to this scheme are required.

Forward Call Indicator (FCI)

- The FCI is a mandatory parameter that is sent in the initial address message (IAM).
- It is used to identify special call capabilities and features.
- It is comprised of two octets of flag bits. Bit M is reserved by American National Standards Institute (ANSI) for use with LNP.
- Bit M indicates that an LNP check has been performed.
- Bit M is updated in the outgoing FCI.

Called Party Number (CPN)

- The Called Party Number is a mandatory IAM parameter that could contain the called party number or the Location Routing Number (LRN). If it contains the LRN and the LNP_PROTOCOL_STREAM is set to UCS08 or later releases, then an FCI with bit M set and the GAP for LNP must also be present.
- The LRN is a 10-digit number (directory number [DN]) that uniquely identifies an end-office to the network for call routing purposes. The LRN is the number the switch uses to route the call.

Generic Address Parameter (GAP)

- The GAP is an optional parameter that is transported in the IAM and contains address digits.
- LNP uses a GAP to store the directory number that has been ported.

  Note: This is the number returned in the SCP **`Analyze_Route`** response message's ***`GenericAddressList`*** with `PortedDialedNo` parameter.

- This usage is denoted by the Ported Dialed Number type of address value (#C0).

Note: More than one GAP can be present in an IAM.

JIP

- The JIP is an optional parameter that is transported in the IAM and contains geographic and service provider information that is associated with the calling party.
- LNP uses this parameter to store the originating switch's location routing number (LRN).
- A default JIP may be provisioned against the originating trunk group through the JIP option in table TRKGRP. This default JIP is used in the outgoing IAM and NetworkBuilder query messages when there is no incoming JIP available.

Note: Refer to table TRKGRP in the *UCS DMS-250 Data Schema Reference Manual* for provisioning the JIP option.

SS7 TCAP messages

SS7 TCAP is used by NetworkBuilder to send messages between the switch and Service Control Point (SCP) or SCP Simulator. NetworkBuilder uses Carrier Advanced Intelligent Network (CAIN) 0.2 TCAP specifications and IN/1 Bellcore specification *TR-NWT-000533* to communicate with SCPs. This switch to SCP and SCP to switch communication occurs through the messages.

LNP occurs during the **Analyze_Information** PIC and uses the LNP **Info_Analyzed** and **Analyze_Route** messages. Refer to *UCS DMS-250 NetworkBuilder Application Guide* for a comprehensive list and details of the NetworkBuilder messages and their parameters.

AIN 0.1 vs. AIN 0.2 Messages

The SSP also needs to communicate with LNP SCPs that are Advanced Intelligent Network (AIN) 0.1-based; therefore, the LNP **Info_Analyzed** message sent by the SSP to the LNP SCP must conform to Illinois Commerce Commission (ICC)/Bellcore AIN 0.1 specifications. Refer to Chapter 6, “Outgoing message parameters,” for more information on the **Info_Analyzed** message.

The most distinguishing changes, from an LNP perspective, for AIN 0.1 messages to AIN 0.2 messages are the following:

- the ability to send extension parameters to the LNP SCP
- support for variations to the ***UserID*** parameter
- new ***TriggerCriteriaType*** parameters
- ***BearerCapability*** restrictions
- restrictions to the nature of address (NOA) and numbering plans for AIN digit type parameters

The SSP must also be set up to handle AIN 0.1-compliant messages received from the LNP SCP. For example, in an AIN 0.1 **Analyze_Route** message, the SSP will not receive a ***ForwardCallIndicator*** or ***GenericAddressListo*** with PortedDialedNo parameter. (These parameters are not received because they are not valid for AIN 0.1 protocol.) If non-AIN 0.1 parameters are sent from the LNP SCP, it is considered a non-fatal application error and the call tries to proceed. Refer to Chapter 7, “Incoming message parameters,” for more information on the **Analyze_Route** message.

The SSP continues to use CAIN 0.2 for CAIN queries.

Call processing

Local Number Portability's (LNP) impact on call processing can be divided into three categories:

- LNP queries – queries launched from the LNP *Office_Code* trigger through the trigger table OFFCCODE
- NetworkBuilder queries – queries launched from a trigger table other than table OFFCCODE (for example, table SPECDIG or table CUSTDP)
- incoming LNP information processing – LNP information received in the signaling system #7 (SS7) initial address message (IAM) from a previous switch

This chapter is divided into three sections, based on the above categories:

- LNP queries
- NetworkBuilder queries
- Incoming LNP information

Each section provides an overview of the category and provides call scenarios within the category.

For clarity, the scenarios in this chapter use the exact parameters' names from the service control point's (SCP) **`Analyze_Route`** message and SS7 IAM message, rather than the abbreviated names used throughout this document. Table 5-1 maps the abbreviations to the parameters' names.

Table 5-1
Mapping of abbreviations to parameters' names

Abbreviation	SCP response parameter name	SS7 IAM parameter name
LNP GAP	***`GenericAddressList`*** parameter containing a `PortedDialedNo`	Generic Address Parameter (GAP)
FCI	***`ForwardCallIndicator`***	Forward Call Indicator (FCI)

LNP queries

This section describes LNP queries. It contains

- an overview of LNP queries
- a description of SCP responses to LNP queries
- a description of protocol control
- figures illustrating LNP queries when the protocol stream equals UCS07
- figures illustrating LNP queries when the protocol stream equals UCS08

Note: Unless a software update changes the LNP_PROTOCOL_STREAM parameter values in table CAINPARM for software releases later than UCS08, UCS07 and UCS08 are the only applicable values.

Overview of LNP queries

For LNP queries, NetworkBuilder provides support for LNP determination. When directory numbers (DNs) in an end-office have been ported, an LNP determination must be made. (Ported DNs are numbers of subscribers who have changed carriers.)

NetworkBuilder uses an LNP technique called the Location Routing Number (LRN) method which uses an in-switch trigger table, OFFCCODE, to perform address screening on origination. This table contains the office codes or numbering plan areas (NPAs) which are considered “open” to LNP because one or more DNs with those office codes or NPAs have already been ported.

If a call is subscribed to NetworkBuilder through the OFFCCODE trigger and the DN is a ten-digit offnet national number, then the switch has the ability to check the OFFCCODE table to determine what action to take based on the given criteria. This action includes querying the LNP SCP to determine whether the DN has been ported. If the DN has been ported, the returned message contains routing information, which allows the call to be routed to its destination.

LNP queries use a global title type called CAIN_OFCD_GT. This global title type is hard-coded for the *Office_Code* trigger; it is not datafillable in any trigger table or parameter. Global title address digits contain the full offnet number and are obtained from the ***CalledPartyID*** parameter of the **Info_Analyzed** message.

Restriction of LNP lookup based on STS

In some instances the NPA–NXX portion of a DN is not enough criteria to determine whether the number belongs to a ported exchange. For example, when an N00 call uses IN/1 to translate the N00 number, a ten-digit number may be returned. This number may be a private number that does not require LNP, but may appear to be from a ported exchange. To resolve this problem, the serving translations scheme (STS) distinguishes between the private versus public exchanges.

Table CAINSTS contains STS values and blocks calls from initiating LNP trigger table (OFFCCODE) lookup. To block LNP lookup on certain STS values, datafill the NO_LNP option against the STS value. Any calls that subscribe to NetworkBuilder and have the *Office_Code* trigger enabled, evaluate table CAINSTS before they evaluate table OFFCCODE. Table 5-2 shows the actions taken based on table CAINSTS evaluation.

Table 5-2
Results of table CAINSTS evaluation

STS in CAINSTS	STS not in CAINSTS	NO_LNP datafilled against STS	NO_LNP option not datafilled	then
X		X		Table OFFCCODE is not evaluated. CAINLNP OM register BLKSYSTS is pegged. CAIN906 log is generated.
X			X	Table OFFCCODE is evaluated and call continues.
	X			Table OFFCCODE is evaluated and call continues.

Info_Analyzed message

NetworkBuilder uses the **`Info_Analyzed`** message to query the LNP SCP. The LNP **`Info_Analyzed`** message uses a subset of the AIN 0.1 protocol. This protocol is described in Chapter 4, “SS7 ISUP and SS7 TCAP messaging.”

The LNP_PARAMETER_SET parameter in table CAINPARM allows you to determine whether a complete set or subset of the LNP **`Info_Analyzed`** parameters are included in the LNP **`Info_Analyzed`** message. You can set the parameter to COMPLETE_SET or MINIMAL_SET. The

LNP queries (continued)

MINIMAL_SET option reduces the number of parameters that are encoded and sent to the SCP, thereby providing a real-time cost savings. Table 5-3 lists parameters supported under each setting.

Table 5-3
Supported Info_Analyzed parameters matrix

Info_Analyzed parameters	COMPLETE_SET	MINIMAL_SET
UserID	X	X
BearerCapability	X	X
CalledPartyID	X	X
TriggerCriteriaType	X	X
CallingPartyID	X	
Carrier	X	
ChargeNumber	X	
ChargePartyStationType	X	
JurisdictionInfo	X	
Note: The ***JurisdictionInfo*** parameter is populated if the CAIN_PROTOCOL_STREAM parameter in table CAINPARM is set to UCS08.		

SCP responses

SCP response messages are handled differently for LNP queries than for NetworkBuilder queries. Note the difference in how these response messages are handled for the two query types. Following is a list of the possible SCP response messages for LNP queries:

- **Analyze_Route**
- **Continue**
- **Disconnect**
- **Send_To_Resource**

Note 1: A **Termination_Notification** may be requested through the **Send_Notification** component with the SCP response message.

Note 2: An **ACG** message may be sent with the SCP response message.

Analyze_Route message

Note: Direct termination routing is not supported.

The LNP_PROTOCOL_STREAM parameter in table CAINPARM provides control of LNP **`Analyze_Route`** message processing. The parameter controls which messaging parameters the switch expects to receive in an **`Analyze_Route`** message. For more information on the LNP_PROTOCOL_STREAM parameter, see "Protocol control" on page 5-8.

Critical call-related information could be lost in the following situation: a NetworkBuilder query to an SCP returns an **`Analyze_Route`** message, then an LNP query to an LNP SCP returns a second **`Analyze_Route`** message. The call-related information in the first **`Analyze_Route`** message is lost. To prevent this loss, certain parameters are preserved from the first **`Analyze_Route`** message. Table 5-4 shows the parameters and sub-parameters preserved from the first **`Analyze_Route`** message.

Table 5-4
Parameters preserved in Analyze_Route message

Parameters preserved in `Analyze_Route` message	sub-parameters
`CalledPartyID`	
`CallingPartyID`	
`ChargeNumber`	
`Carrier`	
`AlternateCarrier`	
`SecondAlternateCarrier`	
`AMADigitsDialedWC`	

—continued—

LNP queries (continued)

Table 5-4
Parameters preserved in Analyze_Route message (continued)

Parameters preserved in `Analyze_Route` message	**sub-parameters**
ExtensionParameter	servTranslationScheme
	reorigAllowed
	satRestriction
	classOfSvc
	Note: Class of service screening is not performed on the LRN that is returned in response to an LNP query.
	cainGroup
	billSequenceNumber
	netinfo
	callBranding
GenericAddressList	OverflowRoutingNo
	DialedNoInwardService
—end—	

Continue message

In a **`Continue`** message for an LNP query, bit M is not set in the FCI of the outgoing signaling system #7 (SS7) initial address message (IAM), which indicates that the SCP was unable to perform the LNP function. Default routing takes place.

Disconnect message

`Disconnect` messages are handled the same for LNP queries as for NetworkBuilder queries; bit M of the Forward Call Indicator (FCI) parameter is not set.

Send_To_Resource message

When a **Send_To_Resource** message in a conversation package is received, NetworkBuilder sends a **Resource_Clear** message with a ***ClearCause*** parameter value of taskRefused and closes the Transaction Capabilities Application Part (TCAP) transaction.

When a response package is received with a **Send_To_Resource** message which includes a disconnect flag, NetworkBuilder attempts to perform default routing using the called party address.

In these situations, no LNP information is sent in an outgoing SS7 IAM.

Interactions with Network_Busy , O_Called_Party_Busy, and O_No_Answer messages

When a busy trigger is encountered after an LNP query, the following actions may take place:

- When the busy trigger results in a NetworkBuilder query, it is considered a NetworkBuilder query following the guidelines as described in the “NetworkBuilder queries” section of this chapter.
- When the busy trigger results in an LNP query, it follows the description in this chapter for LNP queries. LNP information from a previous LNP query, if any, is discarded when an **Analyze_Route** message is received in response to the busy trigger query.
- Normally, a busy trigger results in alternate route choices from the previous **Analyze_Route** message being used. If the previous **Analyze_Route** was in response to an LNP query, then it is unable to contain any direct routes (for example, ***PrimaryTrunkGroup***) or alternate standard routes (for example OverflowRoutingNo) because only the ***CalledPartyID***, ***GenericAddressList***, and ***ForwardCallIndicator*** parameters are supported for the LNP **Analyze_Route** message. For this scenario, alternate routing proceeds as follows:
  - — Exhaust the standard route list (in the case of NETBUSY).
  - — After the standard routes are exhausted, NetworkBuilder route advancing is performed.

Note: The ***O_No_Answer*** trigger exhibits similar behavior as the busy triggers.

LNP queries (end)

Protocol control

The LNP protocol is controlled by in-switch datafill. LNP protocol control consists of two parts:

- The version determines how LNP parameters are encoded.
- The stream determines which parameters the switch expects to receive from the SCP during an LNP query.

The version is controlled through the LNP_PROTOCOL_VERSION parameter in table CAINPARM. The only accepted value for the LNP_PROTOCOL_VERSION parameter is V1.

The stream is controlled through the LNP_PROTOCOL_STREAM parameter in table CAINPARM. The accepted values for the LNP_PROTOCOL_STREAM parameter are UCS07 and UCS08. Table 5-5 shows the difference between the parameters the switch expects to receive when the LNP_PROTOCOL_PARAMETER equals UCS07 and when it equals UCS08.

Note: Unless a software update changes the LNP_PROTOCOL_STREAM parameter values in table CAINPARM for software releases later than UCS08, UCS07 and UCS08 are the only applicable values.

Table 5-5
Differences between LNP_PROTOCOL_STREAM parameter's values

Parameters switch expects	UCS07 setting	UCS08 or later setting
`CalledPartyID` parameter	X	X
`GenericAddressList` parameter containing a `PortedDialedNo`		X
`ForwardCallIndicator` parameter with bit M		X
Note: If the switch receives additional parameters, it ignores them.		

LNP_PROTOCOL_STREAM provisioned as UCS07

When the LNP_PROTOCOL_STREAM parameter is provisioned as UCS07, the switch expects to receive only the ***`CalledPartyID`*** parameter and ignores other parameters.

The five scenarios in this section depict how the switch and the SCP process parameters when the LNP_PROTOCOL_STREAM parameter is set to UCS07:

- Scenario 1: LNP query with LRN returned
- Scenario 2: LNP query with home LRN returned
- Scenario 3: LNP query without an LRN returned
- Scenario 4: LNP query without a ***`CalledPartyID`*** returned
- Scenario 5: Non-ISUP termination and SIGPTDNO

Each of these scenarios contain

- a brief description of the scenario
- step-by-step details of the scenario
- operational measurements (OM) information
- call detail report (CDR) summary

Note: Some of the scenarios contain terminating agent information.

LNP queries
LNP_PROTOCOL_STREAM=UCS07 (continued)

Scenario 1: LNP query with LRN returned

The ***CalledPartyID*** parameter's value from the SCP in an **Analyze_Route** response message is compared to the directory number (DN). If the values are different, the value of the ***CalledPartyID*** parameter returned in the **Analyze_Route** message is treated as the LRN.

Figure 5-1 illustrates Scenario 1. The following explains, in detail, Scenario 1:

1 The switch sends an **Info_Analyzed** message to the SCP. This message contains a ***CalledPartyID*** parameter. The ***CalledPartyID*** parameter contains the directory number (DN).

   Note: Sometimes the switch alters the DN during translations, so the number it sends in the ***CalledPartyID*** parameter is a translated version of the DN. This chapter refers to this number (even if the switch altered it) as the directory number and DN.

2 The SCP responds with an **Analyze_Route** response message. The **Analyze_Route** message contains

   — the trunk group (the value in the ***PrimaryTrunkGroup*** parameter) to use to route the call

   — the LRN within the ***CalledPartyID*** parameter

   Because the DN is a ported number, the ***CalledPartyID*** parameter the SCP returned contains a called party address that is different from the DN. Since, the called party address is different from the DN, the called party address is an LRN.

3 The switch sends an initial address message (IAM) to the next switch. The IAM message contains

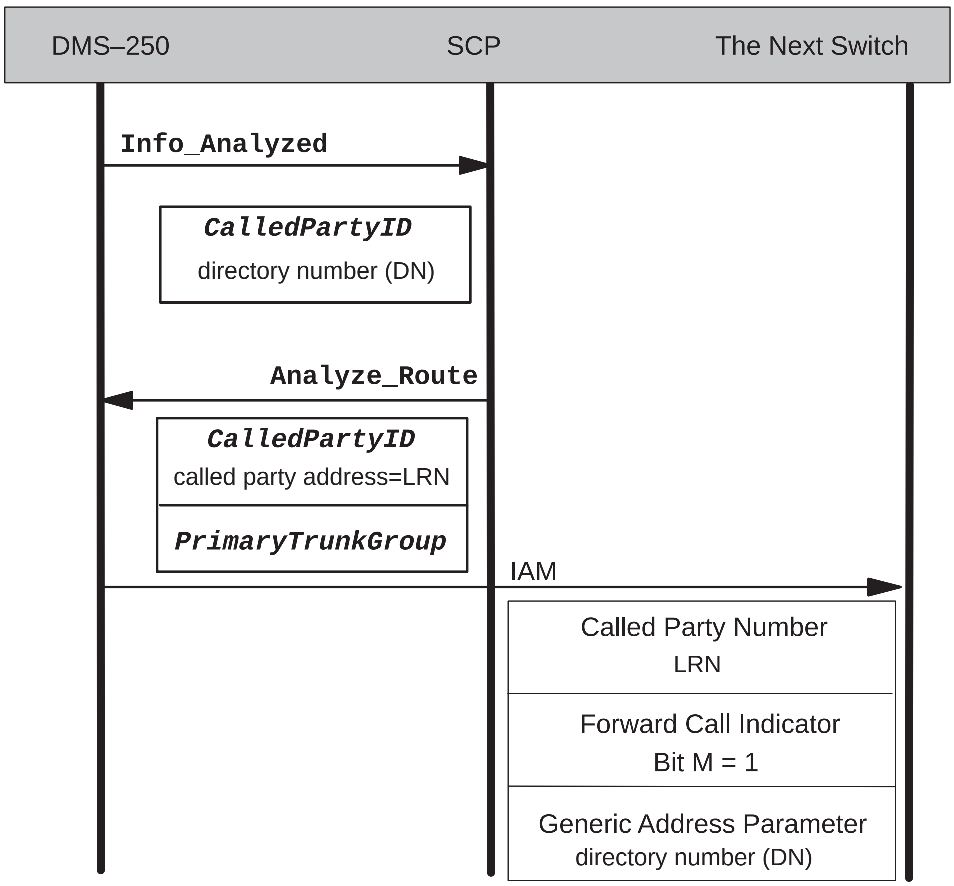
   — the LRN within the Called Party Number (CPN) parameter

   — bit M set within the Forward Call Indicator (FCI) parameter

   — the DN within the Generic Address Parameter (GAP)

LNP queries
LNP_PROTOCOL_STREAM=UCS07 (continued)

Figure 5-1
Scenario 1: LNP query with LRN returned



Note: DN is unequal to the called party address, so the called party address is the LRN.

LNP queries
LNP_PROTOCOL_STREAM=UCS07 (continued)

Terminating agent information

If the terminating agent is an SS7 agent, the switch uses the LRN to route the call and

- sets bit M in the Forward Call Indicator (FCI) parameter of the outgoing IAM
- places the directory number (DN) in the Generic Address Parameter (GAP) of the outgoing IAM
- places the LRN in the Called Party Number (CPN) of the outgoing IAM

If the terminating agent is a non-SS7 agent, the called party address in the query message is outpulsed and is used to route the call.

OM information

Registers OFCDLOOK, LNPQUERY, and LRNONLNP are pegged in OM group CAINLNP.

CDR summary information

Table 5-6 summarizes the CDR population for Scenario 1.

Table 5-6
Scenario 1 CDR summary

LNPCHECK	DIALEDNO	CALLEDNO	OUTPULNO
5	incoming IAM’s CPN	query message’s ***CalledPartyID***	query message’s ***CalledPartyID***
TERMLRN	**PORTEDNO**	**ORIGLRN**	**CN1REQ**
empty	empty	empty	9

Scenario 2: LNP query with home LRN returned

The switch receives an LRN from the SCP and determines that the LRN is its home LRN.

Figure 5-2 illustrates Scenario 2. The following explains, in detail, Scenario 2:

1 The switch sends an **`Info_Analyzed`** message to the SCP. This message contains a ***`CalledPartyID`*** parameter. The ***`CalledPartyID`*** parameter contains the directory number (DN).

2 The SCP responds with an **`Analyze_Route`** response message. The **`Analyze_Route`** message contains the LRN within the ***`CalledPartyID`*** parameter.

   Because the DN is a ported number, the ***`CalledPartyID`*** parameter returned by the SCP contains a called party address that is different from the DN. Since, the called party address is different from the DN, the called party address is an LRN.

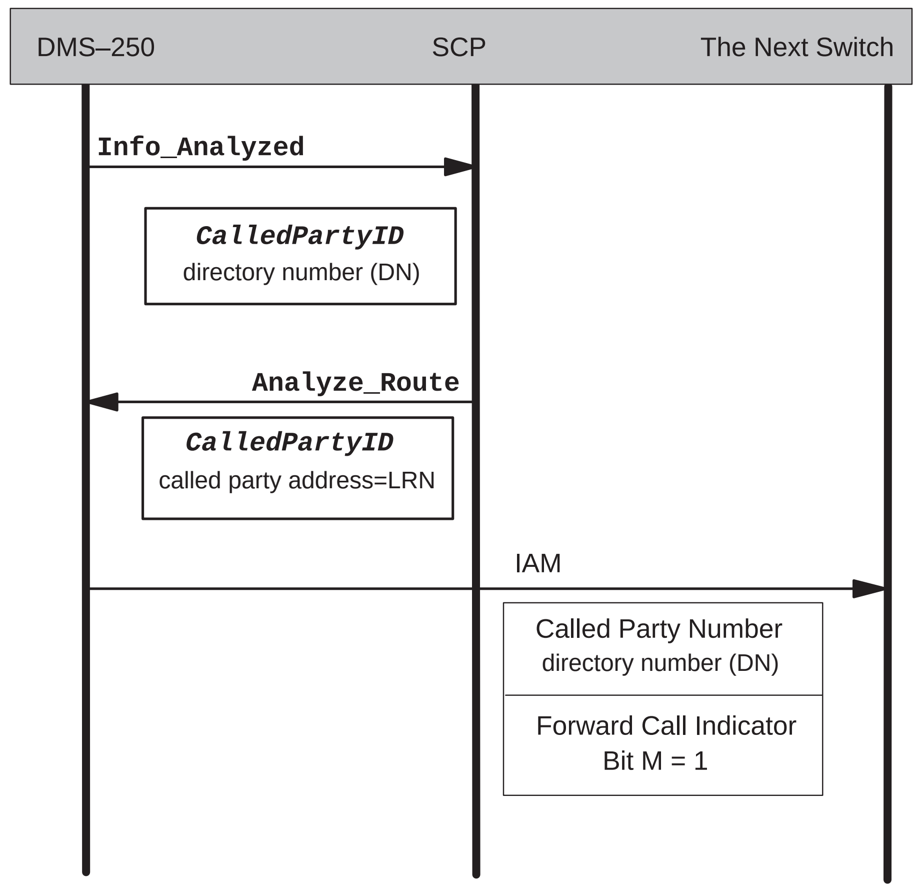
3 The switch determines that the LRN is one of its home LRNs.

   Note: To determine whether the LRN is one of its home LRNs, the switch verifies that the LRN is datafilled within table TERMLRN.

4 The switch performs translations on the directory number (DN) and routes the call.

LNP queries
LNP_PROTOCOL_STREAM=UCS07 (continued)

Figure 5-2
Scenario 2: LNP query with home LRN returned



Note: DN is unequal to the called party address, so the called party address is the LRN.

Terminating agent information

If the terminating agent is an SS7 agent, the switch

- sets bit M of the Forward Call Indicator (FCI) parameter in the outgoing IAM
- places the directory number (DN) in the Called Party Number (CPN) parameter of the outgoing IAM

Note: The switch does not send LNP information in a Generic Address Parameter (GAP) of the outgoing IAM.

If the terminating agent is a non-SS7 agent, the switch outpulses the directory number (DN).

OM information

Registers OFCDLOOK, LRNONLNP, LNPQUERY, and TERMLRN in the OM Group CAINLNP are pegged.

CDR summary information

Table 5-7 summarizes the CDR population for Scenario 2.

Table 5-7
Scenario 2 CDR Summary

DIALEDNO	CALLEDNO	OUTPULNO	TERMLRN
incoming IAM’s CPN	query message’s ***`CalledPartyID`***	query message’s ***`CalledPartyID`***	SCP response’s LRN
LNPCHECK	**ORIGLRN**	**PORTEDNO**	**CN1REQ**
7	JIP (if present) in IAM	empty	9

LNP queries
LNP_PROTOCOL_STREAM=UCS07 (continued)

Scenario 3: LNP query without an LRN returned

The ***CalledPartyID*** parameter's value from the SCP in an **Analyze_Route** response message is compared to the DN. If the values are equal, the SCP has not returned an LRN.

Figure 5-3 illustrates Scenario 3. The following explains, in detail, Scenario 3:

1 The switch sends an **Info_Analyzed** message to the SCP. This message contains a ***CalledPartyID*** parameter. The ***CalledPartyID*** parameter contains the directory number (DN).

2 The SCP responds with an **Analyze_Route** response message. The **Analyze_Route** message contains

— the trunk group (the value in the ***PrimaryTrunkGroup*** parameter) to use to route the call

— the called party address within the ***CalledPartyID*** parameter

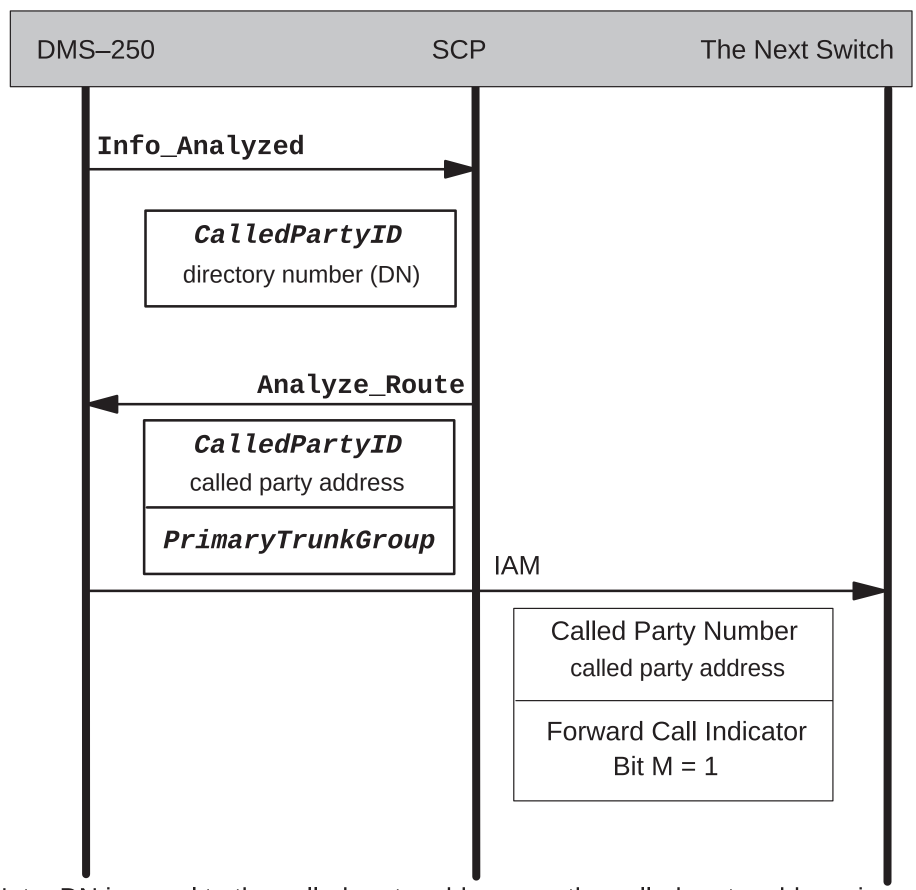
Because the ***CalledPartyID*** parameter's value equals the the DN's value, the***CalledPartyID*** parameter's value is not an LRN.

3 The switch sends an IAM to the next switch. The IAM contains the ***CalledPartyID*** parameter's value in the Called Party Number (CPN) parameter.

Note: If the **Analyze_Route** response message returns an invalid address or an address of an incorrect length, the switch also uses the value within the ***PrimaryTrunkGroup*** parameter to route the call.

LNP queries
LNP_PROTOCOL_STREAM=UCS07 (continued)

Figure 5-3
Scenario 3: LNP query without an LRN returned



Note: DN is equal to the called party address, so the called party address is not the LRN.

LNP queries
LNP_PROTOCOL_STREAM=UCS07 (continued)

Terminating agent information

If the terminating agent is an SS7 agent, the switch

- sets bit M of the Forward Call Indicator (FCI) parameter in the outgoing IAM
- places the called party address(the value returned in the ***CalledPartyID*** parameter) into the Called Party Number (CPN) parameter of the outgoing IAM

Note: The switch does not send LNP information in a Generic Address Parameter (GAP) of the outgoing IAM.

If the terminating agent is a non-SS7 agent, the switch uses the called party address to route the call.

OM information

Registers OFCDLOOK and LNPQUERY are pegged in OM group CAINLNP.

CDR summary

Table 5-8 summarizes the CDR population for Scenario 3.

Table 5-8
Scenario 3 CDR summary

LNPCHECK	DIALEDNO	CALLEDNO	OUTPULNO
5	incoming IAM’s CPN	incoming IAM’s CPN	incoming IAM’s CPN
TERMLRN	**PORTEDNO**	**CN1REQ**	
empty	empty	9	

Scenario 4: LNP query without a CalledPartyID returned

If the **Analyze_Route** message is returned without a ***CalledPartyID*** parameter, the switch determines that the SCP did not return an LRN and uses default routing to route the call.

Figure 5-4 illustrates Scenario 4. The following explains, in detail, Scenario 4:

1 The switch sends an **Info_Analyzed** message to the SCP.

2 The SCP returns an **Analyze_Route** message. The **Analyze_Route** message does not contain the ***CalledPartyID*** parameter, but does contain

  — the trunk group (the value in the ***PrimaryTrunkGroup*** parameter) to use to route the call

  — a ***ForwardCallIndicator*** parameter with bit M set

3 The switch determines that the SCP did not return an LRN.

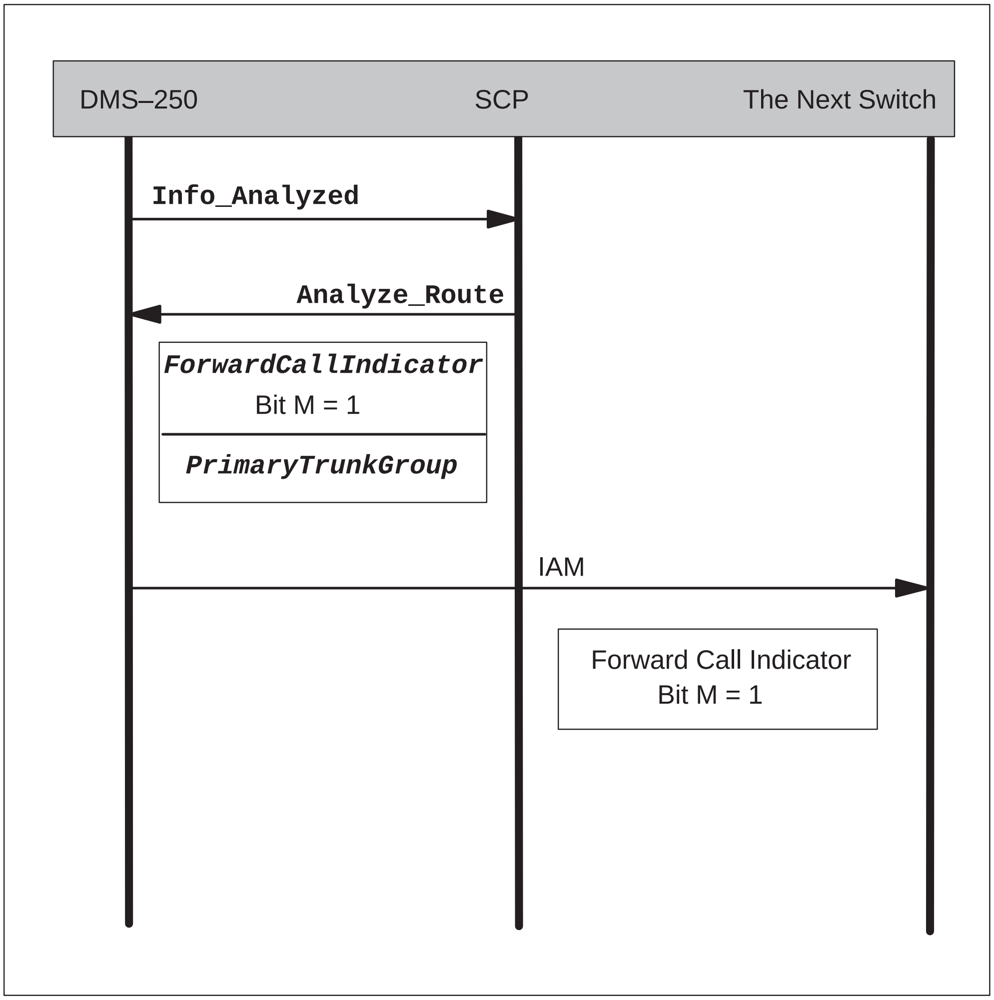
4 The switch

  — uses default routing to route the call

  — sets bit M in the Forward Call Indicator (FCI) parameter of the outgoing IAM

LNP queries
LNP_PROTOCOL_STREAM=UCS07 (continued)

Figure 5-4
Scenario 4: LNP query without a CalledPartyId returned



OM information

Registers OFCDLOOK and LNPQUERY are pegged in OM group CAINLNP.

CDR summary

Table 5-9 summarizes the CDR population for Scenario 4.

Table 5-9
Scenario 4 CDR summary

LNPCHECK	DIALEDNO	CALLEDNO	OUTPULNO
5	incoming IAM’s CPN	incoming IAM’s CPN	incoming IAM’s CPN
TERMLRN	**PORTEDNO**	**CN1REQ**	
empty	empty	9	

LNP queries
LNP_PROTOCOL_STREAM=UCS07 (continued)

Scenario 5: Non-ISUP termination and SIGPTDNO

If the terminating agent is an SS7 agent, the switch builds and sends the LNP information as part of the IAM message. However, the LNP information is discarded if the SIGPTDNO option is datafilled against the trunk in table TRKGRP (for SS7 IMT or SS7 FGD trunks) or table TRKFEAT (for AXXESS trunks).

Figure 5-5 illustrates Scenario 5. The following explains, in detail, Scenario 5:

1 The switch sends an **`Info_Analyzed`** message to the SCP.

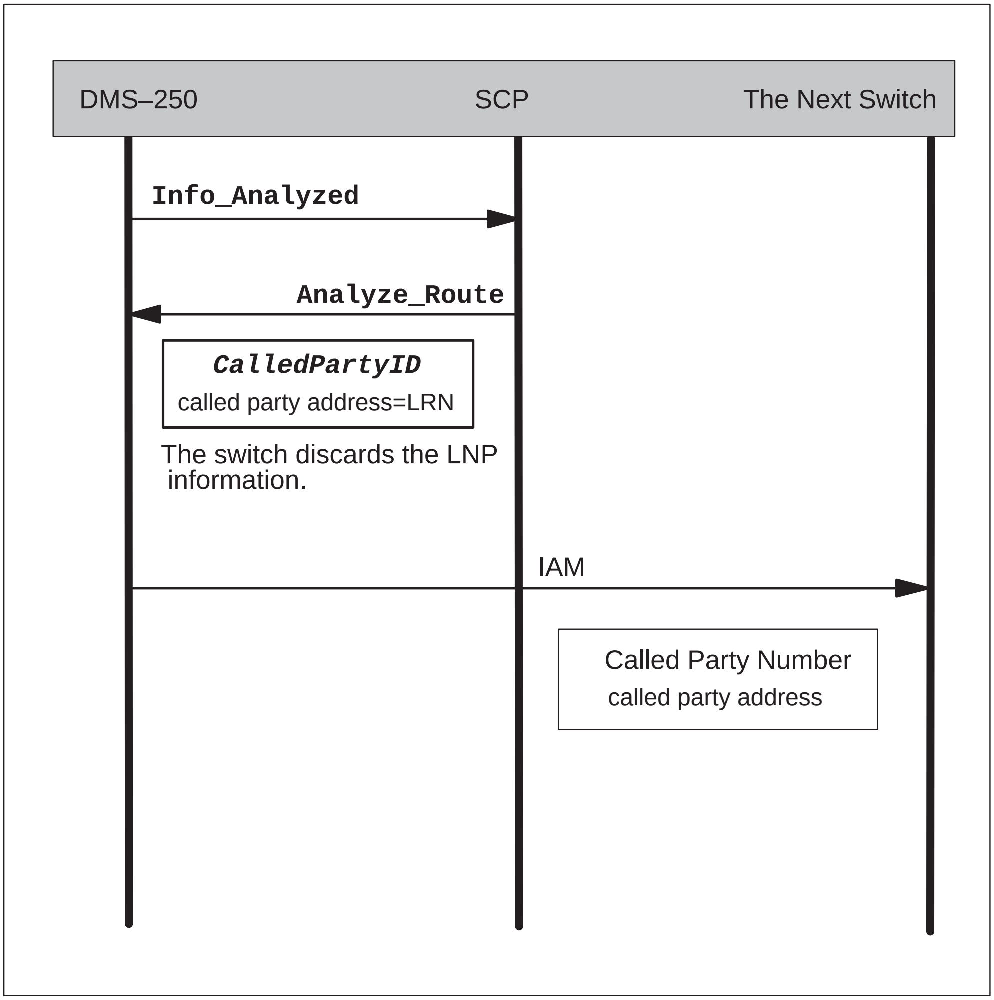
2 The SCP returns an **`Analyze_Route`** message.

3 The switch
   - — discards the LNP information
   - — does not send LNP information in a Generic Address Parameter (GAP) of the outgoing IAM
   - — does not set bit M in the Forward Call Indicator (FCI) parameter of the outgoing IAM

4 The switch replaces the LRN in the IAM's Called Party Number (CPN) parameter with the called party address.

Note: If the terminating agent is a PTS agent, the above information also applies.

Figure 5-5
Scenario 5: Non-ISUP termination



LNP queries
LNP_PROTOCOL_STREAM=UCS07 (end)

OM information

The LNPDISCD register is pegged in OM group CAINLNP.

CDR summary

Table 5-10 summarizes the CDR population for Scenario 5.

Table 5-10
Scenario 5 CDR summary

LNPCHECK	DIALEDNO	CALLEDNO	OUTPULNO
5	incoming IAM’s CPN	incoming IAM’s CPN	incoming IAM’s CPN
TERMLRN	**PORTEDNO**	**CN1REQ**	
empty	empty	9	

LNP_PROTOCOL_STREAM provisioned as UCS08

Note: Unless a software update changes the LNP_PROTOCOL_STREAM parameter values in table CAINPARM for software releases later than UCS08, UCS07 and UCS08 are the only applicable values.

When the LNP_PROTOCOL_STREAM parameter is provisioned as UCS08, the switch expects to receive

- the ***`CalledPartyID`*** parameter
- a ***`GenericAddressList`*** parameter containing a `PortedDialedNo`
- the ***`ForwardCallIndicator`*** parameter with bit M set

For clarity, the scenarios in this chapter use the exact parameters' names from the SCP **`Analyze_Route`** message and SS7 IAM message, rather than the abbreviated names used throughout this document. Table 5-11 maps the abbreviation to the parameters' names.

Table 5-11
Mapping of abbreviations to parameters' names

Abbreviation	SCP response parameter name	SS7 IAM parameter name
LNP GAP	***`GenericAddressList`*** parameter containing a `PortedDialedNo`	Generic Address Parameter (GAP)
FCI	***`ForwardCallIndicator`***	Forward Call Indicator (FCI)

LNP queries
LNP_PROTOCOL_STREAM=UCS08 (continued)

The ten scenarios in this section depict how the switch and the SCP process parameters when the LNP_PROTOCOL_STREAM parameter equals UCS08:

- Scenario 1: LNP query without an LRN returned
- Scenario 2: CalledPartyID, GenericAddressList, and ForwardCallIndicator returned
- Scenario 3: LNP query returning home LRN
- Scenario 4: LNP query without an LRN returned
- Scenario 5: CalledPartyID, no GenericAddressList, but ForwardCallIndicator returned
- Scenario 6: CalledPartyID, no GenericAddressList, no ForwardCallIndicator returned
- Scenario 7: No CalledPartyID, GenericAddressList present, ForwardCallIndicator with bit M set returned
- Scenario 8: No CalledPartyId, GenericAddressList present, no ForwardCallIndicator returned
- Scenario 9: No CalledPartyID, no GenericAddressList, ForwardCallIndicator with bit M set returned
- Scenario 10: No CalledPartyID, no GenericAddressList, no ForwardCallIndicator returned

Each of these scenarios contain

- a brief description of the scenario
- step-by-step details of the scenario
- call detail report (CDR) summary

Note: Some of the scenarios contain operational measurements (OM) and terminating agent information.

LNP queries
LNP_PROTOCOL_STREAM=UCS08 (continued)

Scenario 1: LNP query without an LRN returned

The ***CalledPartyID*** parameter's value in an **Analyze_Route** response message is compared to the DN. If the values are equal, the switch determines the SCP has not returned an LRN.

Figure 5-6 illustrates Scenario 1. The following explains, in detail, Scenario 1:

1 The switch sends an **Info_Analyzed** message to the SCP. This message contains a ***CalledPartyID*** parameter. The ***CalledPartyID*** parameter contains the directory number (DN).

2 The SCP responds with an **Analyze_Route** response message. The **Analyze_Route** message contains
   - the trunk group (the value in the ***PrimaryTrunkGroup*** parameter) to use to route the call
   - the called party address within the ***CalledPartyID*** parameter

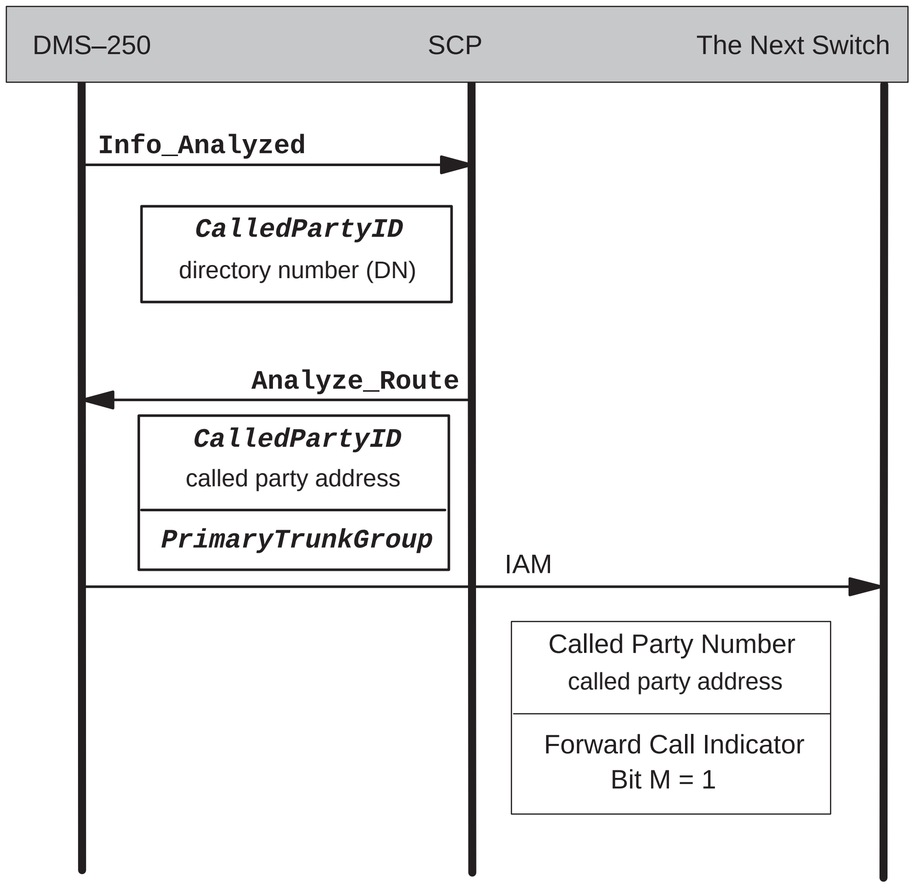
   Because the ***CalledPartyID*** parameter's value equals the the DN's value, the ***CalledPartyID*** parameter's value is not an LRN.

3 The switch sends an IAM to the next switch. The IAM contains
   - the ***CalledPartyID*** parameter's value in the Called Party Number (CPN) parameter
   - the Forward Call Indicator (FCI) parameter with bit M set

Note: If an invalid address or an address of an incorrect length is returned, the switch also uses the value within the ***PrimaryTrunkGroup*** parameter to route the call.

LNP queries
LNP_PROTOCOL_STREAM=UCS08 (continued)

Figure 5-6
Scenario 1: LNP query without an LRN returned



Note: DN is equal to the called party address, so the called party address is not the LRN.

OM information

Registers OFCDLOOK and LNPQUERY are pegged in the OM group CAINLNP.

CDR summary

Table 5-12 summarizes the CDR population for Scenario 1.

Table 5-12
Scenario 1 CDR summary

LNPCHECK	OUTPULNO	DIALEDNO	CALLEDNO
5	incoming IAM's CPN	incoming IAM's CPN	incoming IAM's CPN
PORTEDNO	**TERMLRN**	**CN1REQ**	
empty	empty	9	

LNP queries
LNP_PROTOCOL_STREAM=UCS08 (continued)

Scenario 2: CalledPartyID, GenericAddressList, and ForwardCallIndicator returned

The switch determines that the SCP has returned an LRN when the SCP responds with an **`Analyze_Route`** response message that contains

- the ***`CalledPartyID`*** parameter
- a ***`GenericAddressList`*** parameter containing a `PortedDialedNo`
- the ***`ForwardCallIndicator`*** parameter with bit M set to 1

Figure 5-7 illustrates Scenario 2. The following explains, in detail, Scenario 2:

1 The switch sends an **`Info_Analyzed`** message to the SCP. This message contains a ***`CalledPartyID`*** parameter. The ***`CalledPartyID`*** parameter contains the directory number (DN).

2 The SCP responds with an **`Analyze_Route`** response message. The **`Analyze_Route`** message contains

— the ***`CalledPartyID`*** parameter

— a ***`GenericAddressList`*** parameter containing a `PortedDialedNo`

— the ***`ForwardCallIndicator`*** parameter with bit M set to 1

3 The switch determines that the SCP has returned an LRN.

4 The switch

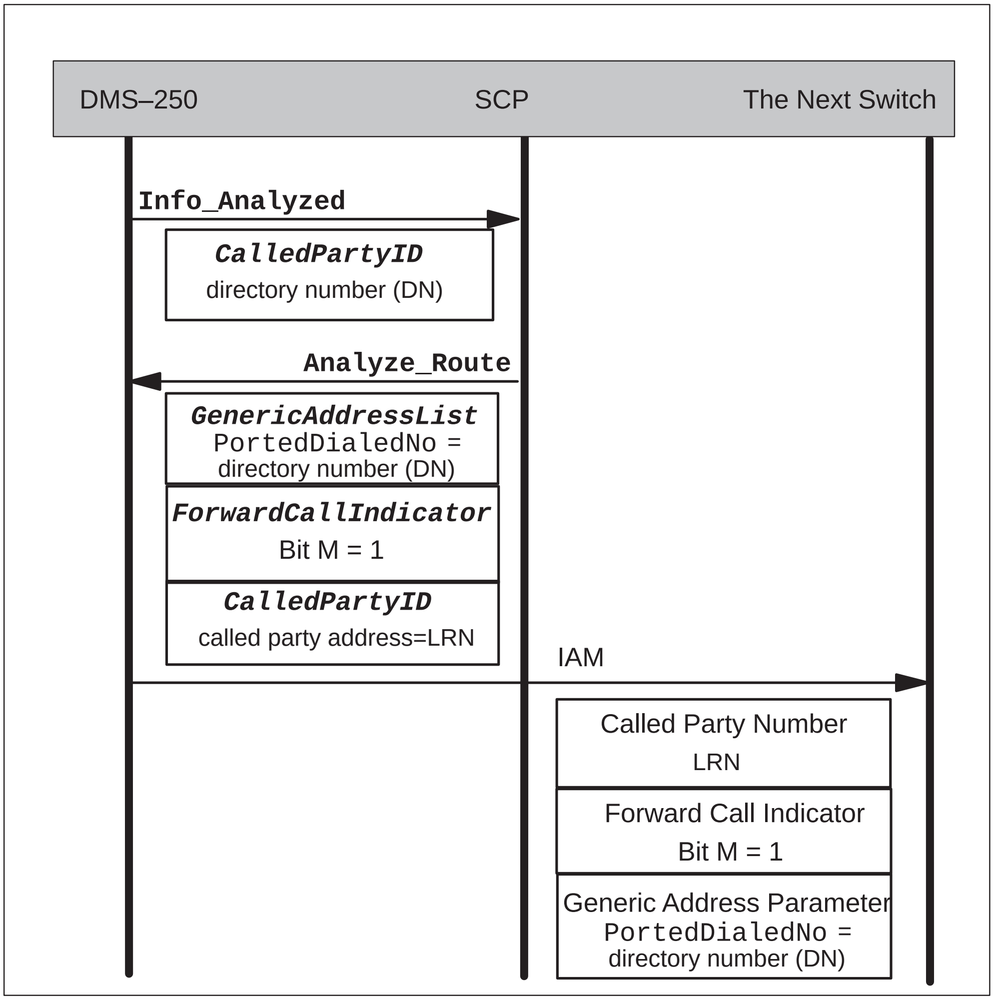
— places the LRN from the ***`CalledPartyID`*** parameter into the IAM's Called Party Number (CPN) parameter

— places the DN in the IAM's Generic Address Parameter (GAP)

— sets bit M of the IAM's Forward Call Indicator (FCI) parameter

LNP queries
LNP_PROTOCOL_STREAM=UCS08 (continued)

Figure 5-7
Scenario 2: CalledPartyID, GenericAddressList, ForwardCallIndicator



LNP queries
LNP_PROTOCOL_STREAM=UCS08 (continued)

OM information

Registers OFCDLOOK, LNPQUERY, and LRNONLNP are pegged.

CDR summary

Table 5-13 summarizes the CDR population for Scenario 2.

Table 5-13
Scenario 2 CDR summary

LNPCHECK	DIALEDNO	CALLEDNO	OUTPULNO
5	incoming IAM’s CPN	SCP response’s LRN	SCP response’s LRN
PORTEDNO			
Original dialed number (DAL and FGD PTS originations) Incoming IAM CPN (SS7 origination)			

Scenario 3: LNP query returning home LRN

The switch receives an LRN from the SCP and determines that the LRN is its home LRN.

Figure 5-8 illustrates Scenario 3. The following explains, in detail, Scenario 3:

1 The switch sends an **`Info_Analyzed`** message to the SCP. This message contains a ***`CalledPartyID`*** parameter. The ***`CalledPartyID`*** parameter contains the directory number (DN).

2 The SCP responds with an **`Analyze_Route`** response message. The **`Analyze_Route`** message contains the LRN within the ***`CalledPartyID`*** parameter.

   Because the DN is a ported number, the ***`CalledPartyID`*** parameter returned by the SCP contains a called party address that is different from the directory number (DN). Since, the called party address is different from the directory number (DN), the called party address is an LRN.

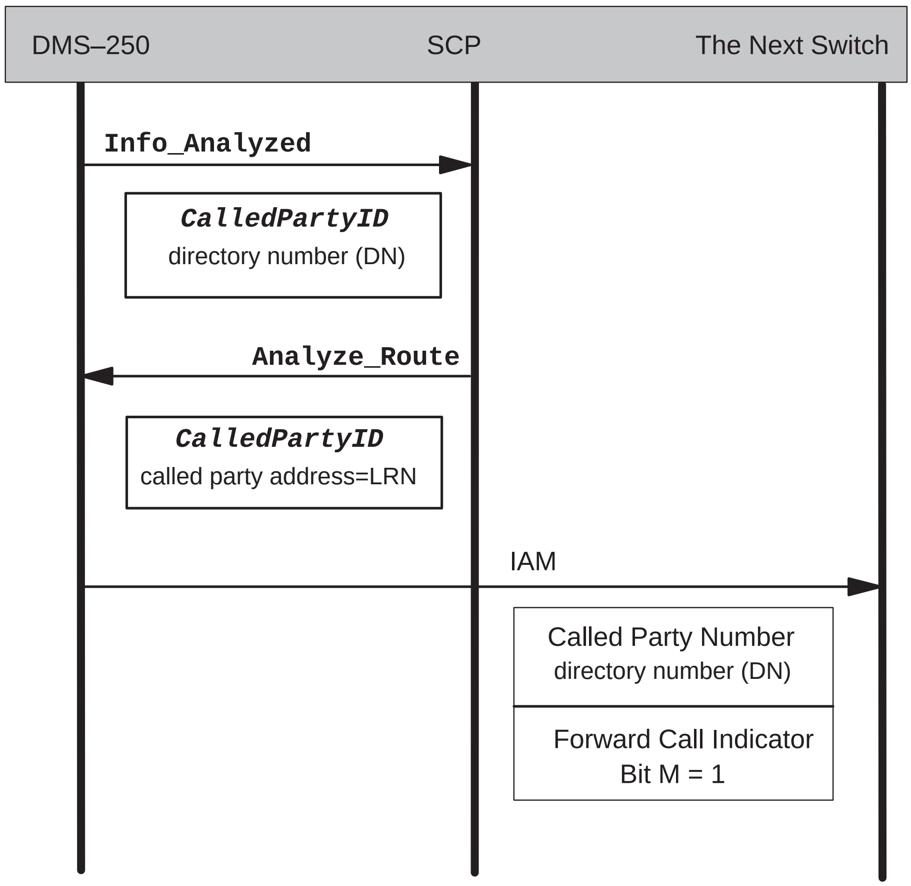
3 The switch determines that the LRN is one of its home LRNs.

   Note: To determine whether the LRN is one of its home LRNs, the switch verifies that the LRN is datafilled within table TERMLRN.

4 The switch performs translations on the DN and routes the call.

LNP queries
LNP_PROTOCOL_STREAM=UCS08 (continued)

Figure 5-8
Scenario 3: LNP query returning home LRN



Note: DN is unequal to the called party address, so the called party address is the LRN.

OM information

The register TERMLRN is pegged.

CDR summary

Table 5-14 summarizes the CDR population for Scenario 3.

Table 5-14
Scenario 3 CDR summary

LNPCHECK	DIALEDNO	OUTPULNO
7	incoming IAM's CPN	query message's ***CalledPartyID***
CALLEDNO	**TERMLRN**	**PORTEDNO**
query message's ***CalledPartyID***	SCP response's LRN	empty
—end—		

LNP queries
LNP_PROTOCOL_STREAM=UCS08 (continued)

Scenario 4: CalledPartyID, GenericAddressList, no ForwardCallIndicator returned

A non-fatal error occurs when an **Analyze_Route** message does not contain the ***ForwardCallIndicator*** parameter, but does contain

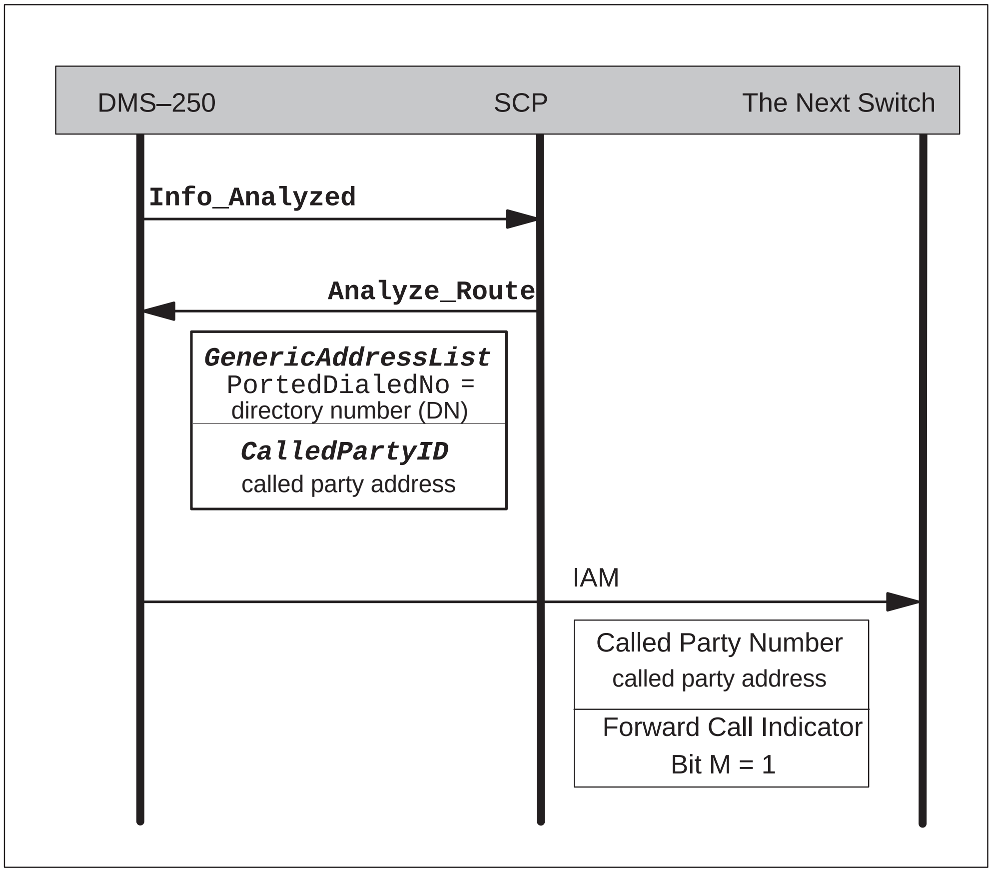
- the ***CalledPartyID*** parameter
- a ***GenericAddressList*** parameter containing a PortedDialedNo

Figure 5-9 illustrates Scenario 4. The following explains, in detail, Scenario 4:

1. The switch sends an **Info_Analyzed** message to the SCP.
2. The SCP does not return the ***ForwardCallIndicator*** parameter in the **Analyze_Route** message, but does return
   - the ***CalledPartyID*** parameter
   - a ***GenericAddressList*** parameter containing a PortedDialedNo
3. The switch produces a CAIN100 log containing the error message "CalledPartyID and LNP GAP but no FCI returned."
4. The switch determines that an LRN has not been returned.
5. The switch
   - sends the called party address (from the ***CalledPartyID*** parameter) in the Called Party Number (CPN) parameter of the outgoing IAM message
   - sets bit M in the Forward Call Indicator (FCI) parameter of the outgoing IAM message

LNP queries
LNP_PROTOCOL_STREAM=UCS08 (continued)

Figure 5-9
Scenario 4: CalledPartyID, GenericAddressList, no ForwardCallIndicator



LNP queries
LNP_PROTOCOL_STREAM=UCS08 (continued)

CDR summary

Table 5-15 summarizes the CDR population for Scenario 4.

Table 5-15
Scenario 4 CDR summary

LNPCHECK	DIALEDNO	CALLEDNO	OUTPULNO
5	incoming IAM’s CPN	SCP response’s ***CalledPartyID***	SCP response’s ***CalledPartyID***
CN1REQ	**PORTEDNO**		
9	incoming IAM’s CPN		

Scenario 5: CalledPartyID, no GenericAddressList, but ForwardCallIndicator returned

The switch determines that the SCP has not returned an LRN when the SCP returns an **`Analyze_Route`** message without a ***`GenericAddressList`*** parameter, but with

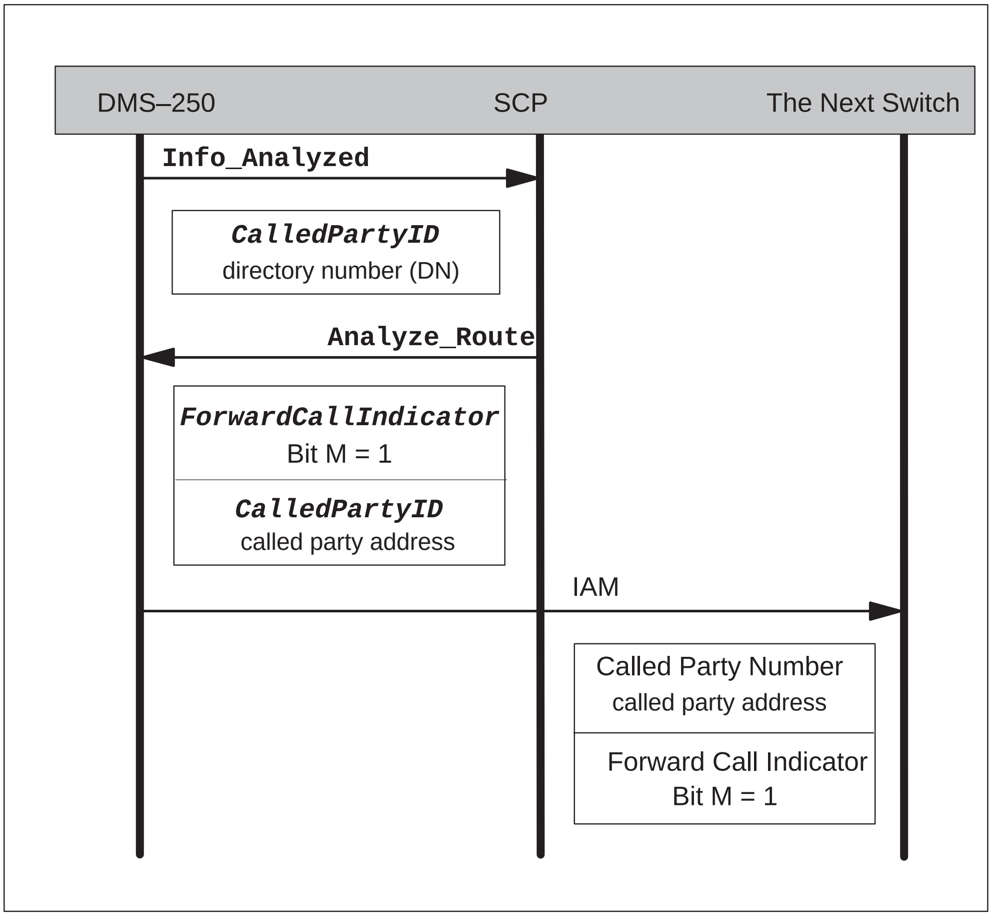
- the ***`CalledPartyID`*** parameter
- the ***`ForwardCallIndicator`*** parameter with bit M set to 1

Figure 5-10 illustrates Scenario 5. The following explains, in detail, Scenario 5:

1. The switch sends an **`Info_Analyzed`** message to the SCP.
2. The SCP returns an **`Analyze_Route`** message without a ***`GenericAddressList`*** parameter, but with
   — a ***`CalledPartyID`*** parameter
   — a ***`ForwardCallIndicator`*** parameter with bit M set to 1
3. The switch determines that the SCP did not return an LRN.
4. The switch sends the called party address from the ***`CalledPartyID`*** parameter in the outgoing IAM’s Called Party Number (CPN) parameter and sets the Forward Call Indicator’s (FCI’s) bit M.

LNP queries
LNP_PROTOCOL_STREAM=UCS08 (continued)

Figure 5-10
Scenario 5: CalledPartyID, no GenericAddressList, ForwardCallIndicator



OM information

The registers OFCDLOOK, LNPQUERY, LRNONLNP are pegged.

CDR summary

Table 5-16 summarizes the CDR population for Scenario 5.

Table 5-16
Scenario 5 CDR summary

DIALEDNO	CALLEDNO	OUTPULNO	PORTEDNO
incoming IAM's CPN	SCP response's ***CalledPartyID***	SCP response's ***CalledPartyID***	empty

LNP queries
LNP_PROTOCOL_STREAM=UCS08 (continued)

Scenario 6: CalledPartyID, no GenericAddressList, no ForwardCallIndicator returned

The switch determines that the SCP has not returned an LRN when the SCP returns an **Analyze_Route** message with a ***CalledPartyID*** parameter, but without the ***GenericAddressList*** and ***ForwardCallIndicator*** parameters.

Figure 5-11 illustrates Scenario 6. The following explains, in detail, Scenario 6:

1 The switch sends an **Info_Analyzed** message to the SCP.

2 The SCP returns an **Analyze_Route** message with a ***CalledPartyID*** parameter, but without the ***GenericAddressList*** and ***ForwardCallIndicator*** parameters.

3 The switch determines that the SCP did not return an LRN.

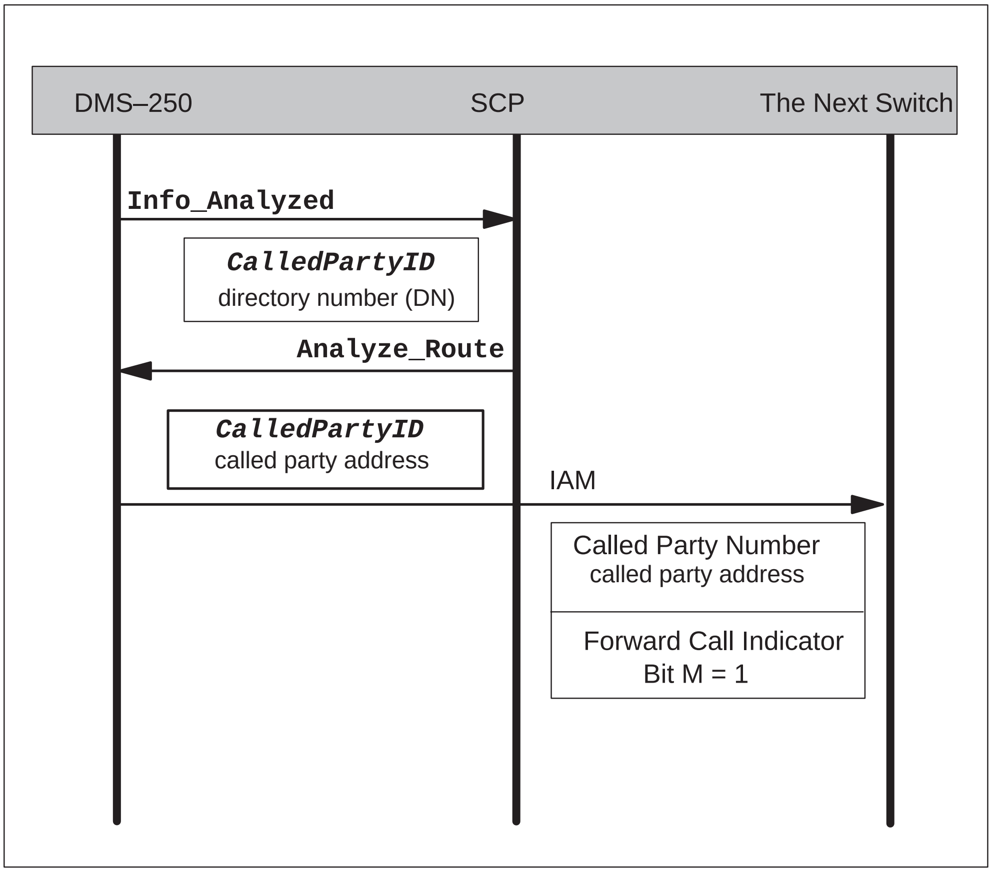
4 The switch

— places the called party address (the value returned in the ***CalledPartyID*** parameter from the **Analyze_Route** message) in the Called Party Number (CPN) parameter of the outgoing IAM message

— sets bit M in the Forward Call Indicator (FCI) parameter of the outgoing IAM message

LNP queries
LNP_PROTOCOL_STREAM=UCS08 (continued)

Figure 5-11
Scenario 6: CalledPartyID, no GenericAddressList, and no ForwardCallIndicator



LNP queries
LNP_PROTOCOL_STREAM=UCS08 (continued)

OM information

The registers OFCDLOOK, LNPQUERY, LRNONLNP are pegged.

CDR summary

Table 5-17 summarizes the CDR population for Scenario 6.

Table 5-17
Scenario 6 CDR summary

LNPCHECK	DIALEDNO	CALLEDNO	OUTPULNO
5	incoming IAM's CPN	SCP response's ***CalledPartyID***	SCP response's ***CalledPartyID***
PORTEDNO	**CN1REQ**		
incoming IAM's CPN	9		

LNP queries
LNP_PROTOCOL_STREAM=UCS08 (continued)

Scenario 7: No CalledPartyID, GenericAddressList present, ForwardCallIndicator with bit M set returned

A non-fatal error occurs when the SCP returns an **Analyze_Route** message without a ***CalledPartyID*** parameter, but with

— the ***ForwardCallIndicator*** parameter

— the ***GenericAddressList*** parameter containing a PortedDialedNo

Figure 5-12 illustrates Scenario 7. The following explains, in detail, Scenario 7:

1 The switch sends an **Info_Analyzed** message to the SCP.

2 The SCP returns an **Analyze_Route** message without a ***CalledPartyID*** parameter, but with

— the ***GenericAddressList*** parameter

— the ***ForwardCallIndicator*** parameter

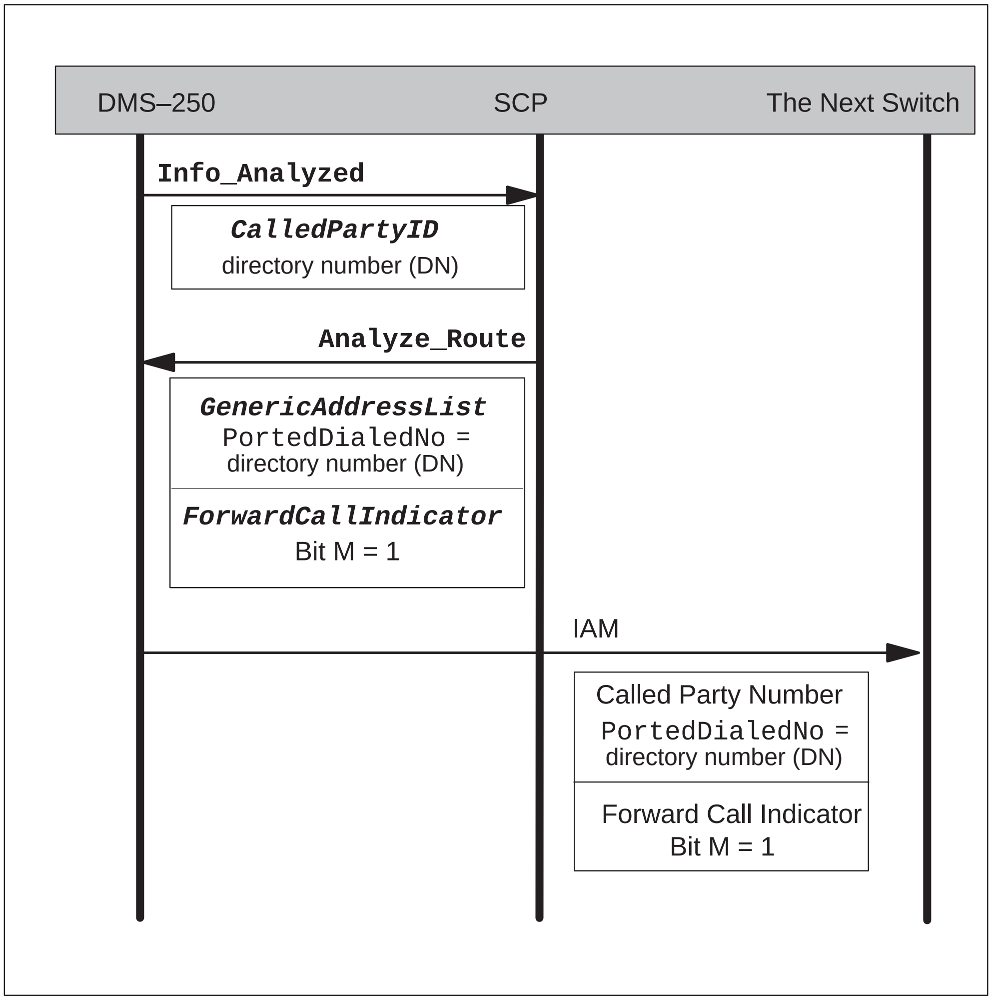
3 The switch produces a CAIN100 log containing the error message “An LNP GAP and FCI but no CalledPartyID returned.”

4 The switch determines that an LRN has not been returned.

5 The switch sends the DN in the outgoing IAM’s Called Party Number (CPN) parameter and sets the Forward Call Indicator (FCI) parameter’s bit M.

LNP queries
LNP_PROTOCOL_STREAM=UCS08 (continued)

Figure 5-12
Scenario 7: No CalledPartyID, GenericAddressList, ForwardCallIndicator



CDR summary

Table 5-18 summarizes the CDR population for Scenario 7.

Table 5-18
Scenario 7 CDR summary

LNPCHECK	DIALEDNO	CALLEDNO	OUTPULNO
5	incoming IAM's CPN	incoming IAM's CPN	incoming IAM's CPN
CN1REQ	**PORTEDNO**		
9	empty		

LNP queries
LNP_PROTOCOL_STREAM=UCS08 (continued)

Scenario 8: No CalledPartyID, GenericAddressList present, no ForwardCallIndicator returned

A non-fatal error occurs when the SCP sends an **`Analyze_Route`** message with a ***`GenericAddressList`*** parameter containing a `PortedDialedNo` without

- the ***`CalledPartyID`*** parameter
- the ***`ForwardCallIndicator`*** parameter

Figure 5-13 illustrates Scenario 8. The following explains, in detail, Scenario 8:

1 The switch sends an **`Info_Analyzed`** message to the SCP.

2 The SCP responds with an **`Analyze_Route`** message. The **`Analyze_Route`** message contains a ***`GenericAddressList`*** parameter containing a `PortedDialedNo`, but does not contain

— the ***`CalledPartyID`*** parameter

— the ***`ForwardCallIndicator`*** parameter

3 The switch produces a CAIN100 log containing the error message “An LNP GAP was returned with no FCI or CalledPartyID.”

4 The switch determines that the SCP did not return an LRN.

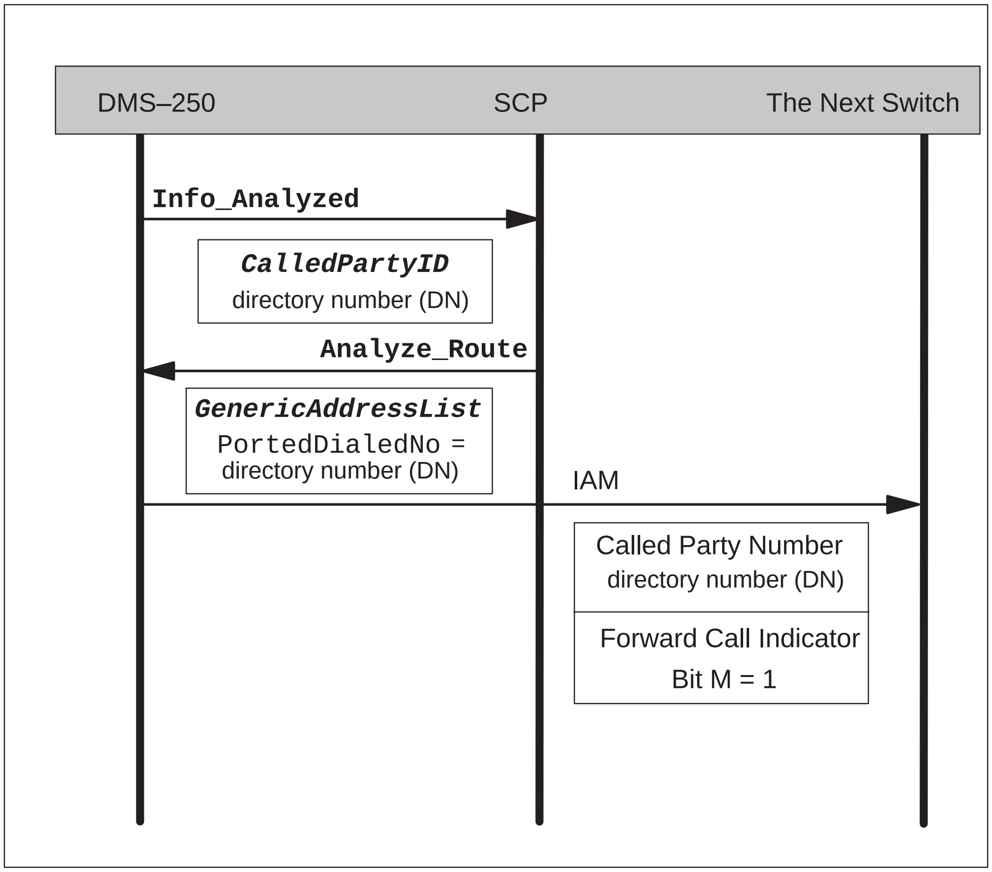
5 The switch

— places the directory number (DN) in the Called Party Number (CPN) parameter of the outgoing IAM message

— sets bit M in the Forward Call Indicator (FCI) parameter of the outgoing IAM message

LNP queries
LNP_PROTOCOL_STREAM=UCS08 (continued)

Figure 5-13
Scenario 8: No CalledPartyID, GenericAddressList, no ForwardCallIndicator



LNP queries
LNP_PROTOCOL_STREAM=UCS08 (continued)

OM information

The registers OFCDLOOK and LNPQUERY are pegged.

CDR summary

Table 5-19 summarizes the CDR population for Scenario 8.

Table 5-19
Scenario 8 CDR summary

LNPCHECK	DIALEDNO	CALLEDNO	OUTPULNO
5	incoming IAM's CPN	incoming IAM's CPN	incoming IAM's CPN
PORTEDNO	**CN1REQ**		
empty	0		

Scenario 9: No CalledPartyID, no GenericAddressList, ForwardCallIndicator with bit M set returned

The switch determines that the SCP did not return an LRN when the SCP responds with an **`Analyze_Route`** message that contains the ***`ForwardCallIndicator`*** parameter, but does not contain

- the ***`CalledPartyID`*** parameter
- a ***`GenericAddressList`*** parameter containing a `PortedDialedNo`

Figure 5-14 illustrates Scenario 9. The following explains, in detail, Scenario 9:

1 The switch sends an **`Info_Analyzed`** message to the SCP.

2 The SCP responds with an **`Analyze_Route`** message. The **`Analyze_Route`** message contains the ***`ForwardCallIndicator`*** parameter, but does not contain

— the ***`CalledPartyID`*** parameter

— a ***`GenericAddressList`*** parameter containing a `PortedDialedNo`

3 The switch determines that the SCP did not return an LRN.

4 The switch

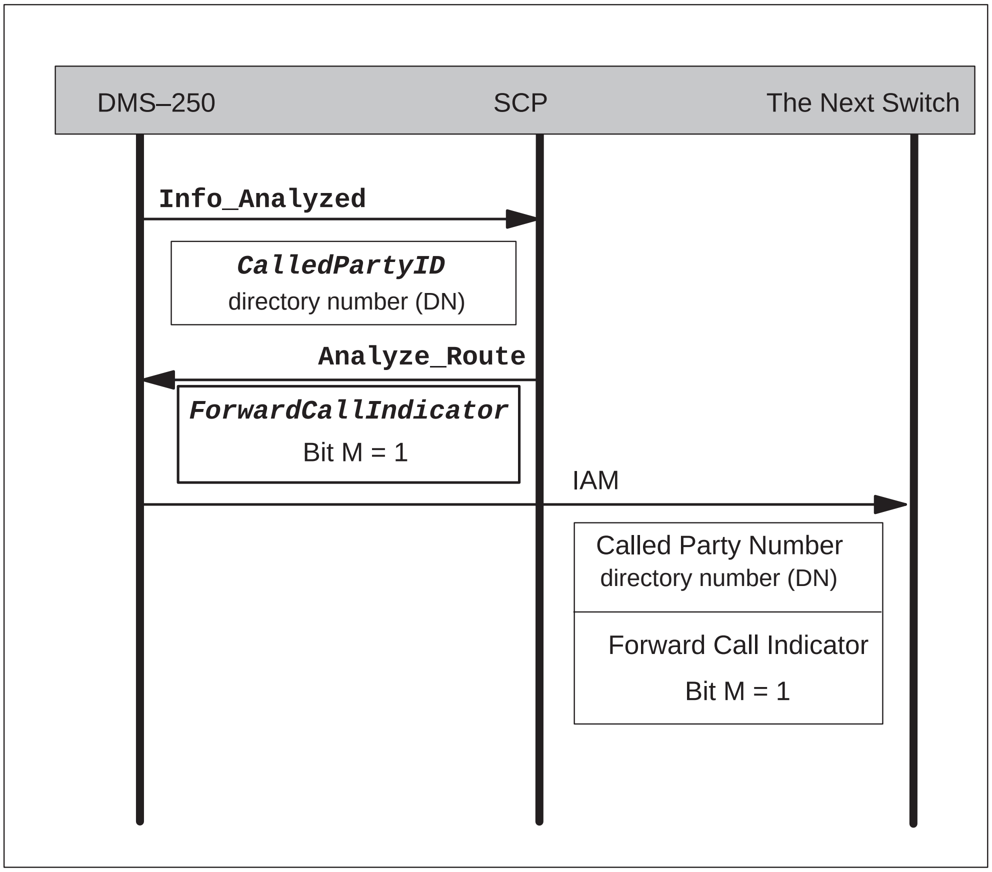
— uses default routing to route the call

— places the directory number (DN) in the Called Party Number (CPN) parameter of the outgoing IAM message

— sets bit M in the Forward Call Indicator (FCI) parameter of the outgoing IAM message

LNP queries
LNP_PROTOCOL_STREAM=UCS08 (continued)

Figure 5-14
Scenario 9: No CalledPartyID, no GenericAddressList, ForwardCallIndicator



OM information

The registers OFCDLOOK and LNPQUERY are pegged.

CDR summary

Table 5-20 summarizes the CDR population for Scenario 9.

Table 5-20
Scenario 9 CDR summary

LNPCHECK	DIALEDNO	CALLEDNO	OUTPULNO
5	incoming IAM’s CPN	incoming IAM’s CPN	incoming IAM’s CPN
CN1REQ	**PORTEDNO**		
0	empty		

LNP queries
LNP_PROTOCOL_STREAM=UCS08 (continued)

Scenario 10: No CalledPartyID, no GenericAddressList, no ForwardCallIndicator returned

The switch determines that the SCP did not return an LRN when the SCP returns an **`Analyze_Route`** message without

- the ***`ForwardCallIndicator`*** parameter
- a ***`GenericAddressList`*** parameter containing a `PortedDialedNo`
- the ***`CalledPartyID`*** parameter

Figure 5-15 illustrates Scenario 10. The following explains, in detail, Scenario 10:

1 The switch sends an **`Info_Analyzed`** message to the SCP.

2 The SCP responds with an **`Analyze_Route`** message. The **`Analyze_Route`** message does not contain

— the ***`CalledPartyID`*** parameter

— the ***`ForwardCallIndicator`*** parameter

— a ***`GenericAddressList`*** parameter containing a `PortedDialedNo`

3 The switch determines that the SCP did not return an LRN.

4 The switch

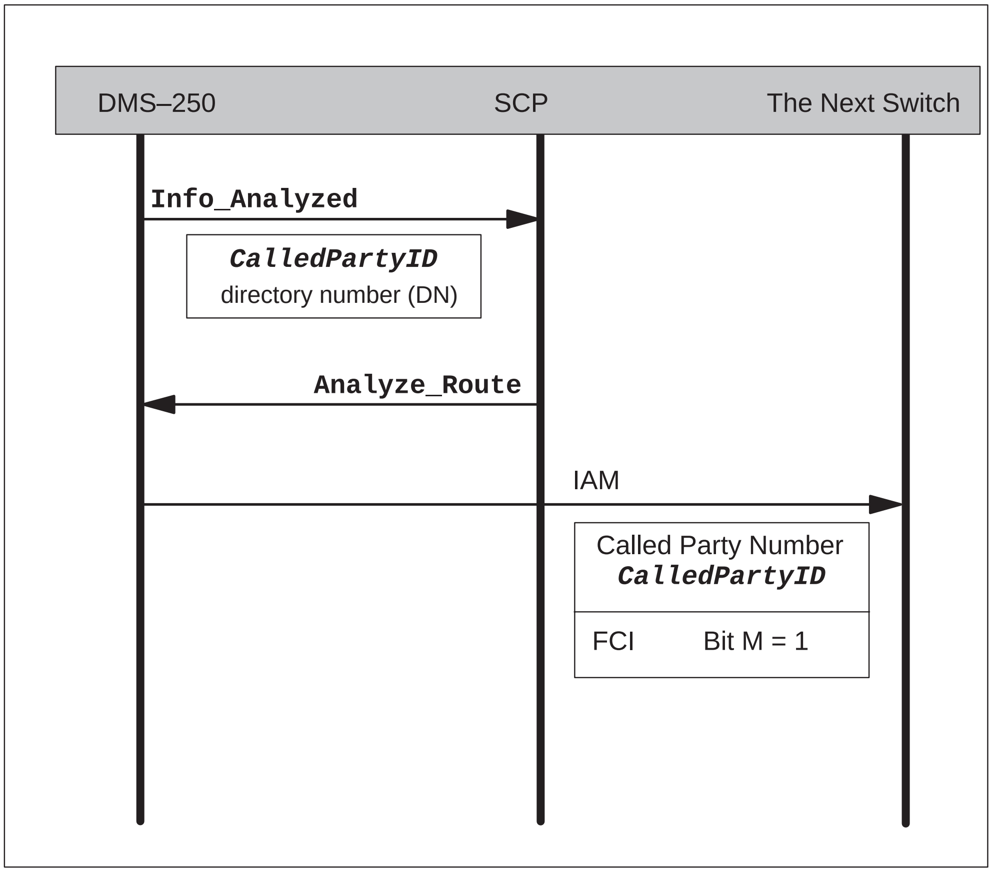
— uses default routing to route the call

— places the directory number (DN) in the Called Party Number (CPN) parameter of the outgoing IAM message

— sets bit M in the Forward Call Indicator (FCI) parameter of the outgoing IAM message

LNP queries
LNP_PROTOCOL_STREAM=UCS08 (continued)

Figure 5-15
Scenario 10: No CalledPartyID, no GenericAddressList, no ForwardCallIndicator



LNP queries
LNP_PROTOCOL_STREAM=UCS08 (end)

OM information

The registers OFCDLOOK and LNPQUERY are pegged.

CDR summary

Table 5-21 summarizes the CDR population for Scenario 10.

Table 5-21
Scenario 10 CDR summary

LNPCHECK	DIALEDNO	CALLEDNO	OUTPULNO
5	incoming IAM's CPN	incoming IAM's CPN	incoming IAM's CPN
CN1REQ	**PORTEDNO**		
0	empty		

NetworkBuilder queries

This section describes NetworkBuilder queries. It contains

- an overview of NetworkBuilder queries
- a description of the differences in the SCP responses to NetworkBuilder queries
- a description of direct termination routing
- a description of standard routing
- figures illustrating NetworkBuilder queries when direct termination routing is used
- figures illustrating NetworkBuilder queries when standard routing is used

The eleven scenarios in this section depict how the switch and the SCP process LNP information during NetworkBuilder queries:

- Scenario 1: GenericAddressList and CalledPartyID returned, ForwardCallIndicator bit M set
- Scenario 2: NetworkBuilder query returning home LRN
- Scenario 3: GenericAddressList and CalledPartyID returned, ForwardCallIndicator bit M not set
- Scenario 4: CalledPartyID returned, ForwardCallIndicator bit M set, no GenericAddressList returned
- Scenario 5: CalledPartyID returned, ForwardCallIndicator bit M not set, no GenericAddressList returned
- Scenario 6: CalledPartyID not returned, GenericAddressList returned, ForwardCallIndicator bit M set
- Scenario 7: CalledPartyID not returned, GenericAddressList returned, ForwardCallIndicator bit M not set
- Scenario 8: CalledPartyID and GenericAddressList not returned, ForwardCallIndicator bit M set
- Scenario 9: CalledPartyID and GenericAddressList not returned, ForwardCallIndicator bit M not set
- Scenario 10: Direct termination and standard routing
- Scenario 11: IN/1 Start message interacting with LNP query

For clarity, the scenarios in this chapter use the exact SCP **`Analyze_Route`** message’s parameters’ names and SS7 IAM message’s parameters’ names, rather than the abbreviated names used throughout this document. Table 5-22 maps the abbreviation to the parameters’ names.

NetworkBuilder queries (continued)

Table 5-22
Mapping of abbreviations to parameters' names

Abbreviation	SCP response parameter name	SS7 IAM parameter name
LNP GAP	***GenericAddressList*** parameter containing a `PortedDialedNo`	Generic Address Parameter (GAP)
FCI	***ForwardCallIndicator***	Forward Call Indicator (FCI)
	—end—	

Overview of NetworkBuilder queries

This section discusses LNP's impact on NetworkBuilder queries. The main impact is not on the queries themselves but on the **Analyze_Route** response message. Support for the ***ForwardCallIndicator*** and ***GenericAddressList*** parameters is added as part of this LNP offering. This means that an LNP-capable switch can accept CAIN 0.2 **Analyze_Route** messages with these LNP parameters and can process these messages accordingly.

Continue message

When the **Continue** message is received, the call continues with call processing as it does for normal NetworkBuilder calls. However, this call could eventually trigger at **Info_Analyzed** for LNP.

Error messages

There is no difference in the processing of the error messages. Error messages include the following:

- **Close**
- **Report_Error**
- **Failure Report**
- **Application_Error**

Info_Analyzed

If an LRN is present and the switch must perform a NetworkBuilder query, then the switch populates the ***CalledPartyID*** parameter with the `PortedDialedNo` from the ***GenericAddressList*** parameter.

Network_Busy, O_Called_Party_Busy, and O_No_Answer messages

When NetworkBuilder calls hit one of the busy triggers and route back to the SCP, the **`Analyze_Route`** message will be handled as discussed in "Direct termination routing" on page 5-60.

If a CAINGRP extension parameter is returned in an **`Analyze_Route`** message, the new CAINGRP is used for the **`Network_Busy`** and **`O_Called_Party_Busy`** triggers and for reorigination at the **`Info_Collected`** and the **`Info_Analyzed`** trigger detection points (TDPs).

The routing functions for **`O_No_Answer`** are handled like **`O_Called_Party_Busy.`**

Connect response message

When a carrier parameter is included in a routing response message that has a value indicating the selected carrier is the local exchange carrier (LEC), call processing resumes at the **`Analyze_Information`** PIC. This allows for reevaluation of the *Office_Code* trigger.

Send_To_Resource and Connect_To_Resource messages

The ***`DestinationAddress`*** parameter does not apply to LNP.

Start message

The switch can initiate a **`Tollfree_Service`** TDP-Request, receive a **`Connect`** response message, and then resume call processing at the **`Analyze_Information`** PIC. This allows for reevaluation of the *Office_Code* trigger.

Other messages

There is no difference in the processing of the following messages: **`Origination_Attempt`**, **`O_Feature_Requested`**, **`Info_Collected`**, **`O_Term_Seized`**, **`O_Answer`**, **`O_Mid_Call`**, **`Timeout`**, **`O_Disconnect`**, **`Resource_Clear`**, **`CTR_Clear`**, **`Call_Info_From_Resource`**, **`Disconnect`**, **`Cancel_Resource_Event`**, **`Collect_Information`**, **`Call_Info_To_Resource`**

NetworkBuilder queries (continued)

Routing

In response to NetworkBuilder queries, the SCP can return direct termination routing or standard routing information.

Direct termination routing

When the switch sent a NetworkBuilder query, rather than an LNP query, to the SCP, the SCP can return direct termination routing information.

The parameters ***`PrimaryTrunkGroup`***, ***`AlternateTrunkGroup`***, and ***`SecondAlternateTrunkGroup`*** identify direct termination route indexes into tables TANDMRTE and TERMRTE.

For NetworkBuilder queries that return LNP information, the SCP only returns the LRN in the ***`CalledPartyID`***. If the SCP returns an `OverflowRoutingNo`, the `OverflowRoutingNo` does not contain the LRN.

Standard routing

In standard routing scenarios, when the SCP returns a ***`CalledPartyID`*** and an `OverflowRoutingNo`, the address digits are evaluated at the **`Info_Analyzed`** TDP, eventually reaching the *Office_Code* trigger.

When the switch uses standard routing (for example, ***`CalledPartyID`***) to route the call and the SCP returns a CAINGRP extension parameter in the **`Analyze_Route`** message, the new CAINGRP is used for the *Office_Code* trigger check. This is also true in cases where the direct termination routes specified in an **`Analyze_Route`** message are busy and standard routing must be tried.

The standard routing parameters require in-switch translations to derive the route index. One route index is calculated for each parameter. The standard routing parameters are the ***`CalledPartyID`*** and/or the `servTranslationScheme` (or `univIdx` for SS7 Global-IMTs), and the ***`GenericAddressList`*** parameter's `OverflowRoutingNo`.

Note: NetworkBuilder handles standard routing by translating and screening the given address. If a route is found, CAIN routes the call based on this address. The address is outpulsed to the next switch unless the terminator is a DAL, or an AXXESS agent mimicking the signaling characteristics of a DAL. The CALLEDNO and OUTPULNO CDR fields are captured. For more information on standard routing, refer to the *UCS DMS-250 NetworkBuilder Application Guide*.

Direct termination and standard routing

When the SCP returns both direct termination and standard routing information in an **`Analyze_Route`** message, the switch first attempts direct routes. If the direct routes are unsuccessful, such as in the busy scenarios, the ***`CalledPartyID`*** and `OverflowRoutingNo` are evaluated at the *Office_Code* trigger. In addition, for both direct termination and standard routing scenarios, reevaluation at the *Office_Code* trigger will not occur if the SCP returns an LRN or an ***`ForwardCallIndicator`*** parameter with bit M set.

See Scenario 10: Direct termination and standard routing on page 5-89 for more information.

Scenario 1: GenericAddressList and CalledPartyID returned, ForwardCallIndicator with bit M set

The switch uses the value in the ***PrimaryTrunkGroup*** parameter to route the call after it has launched a NetworkBuilder query and the SCP has responded with an **Analyze_Route** message containing

- the ***CalledPartyID*** parameter
- the ***GenericAddressList*** parameter
- the ***ForwardCallIndicator*** parameter's bit M set to 1

Note: For this scenario, avoid use of the ***OutpulseNumber*** field. Use of the ***OutpulseNumber*** field for routing could cause erroneous results.

Figure 5-16 illustrates Scenario 1. The following explains, in detail, Scenario 1:

1 The switch launches a NetworkBuilder query.

2 The SCP responds with an **Analyze_Route** message containing

   — the ***CalledPartyID*** parameter

   — the ***GenericAddressList*** parameter containing a `PortedDialedNo`

   — the ***ForwardCallIndicator*** parameter's bit M set to 1

   — the ***PrimaryTrunkGroup*** parameter

3 The switch uses the value in the ***PrimaryTrunkGroup*** parameter to route the call and

   — places the `PortedDialedNo` in the outgoing IAM's Generic Address Parameter (GAP)

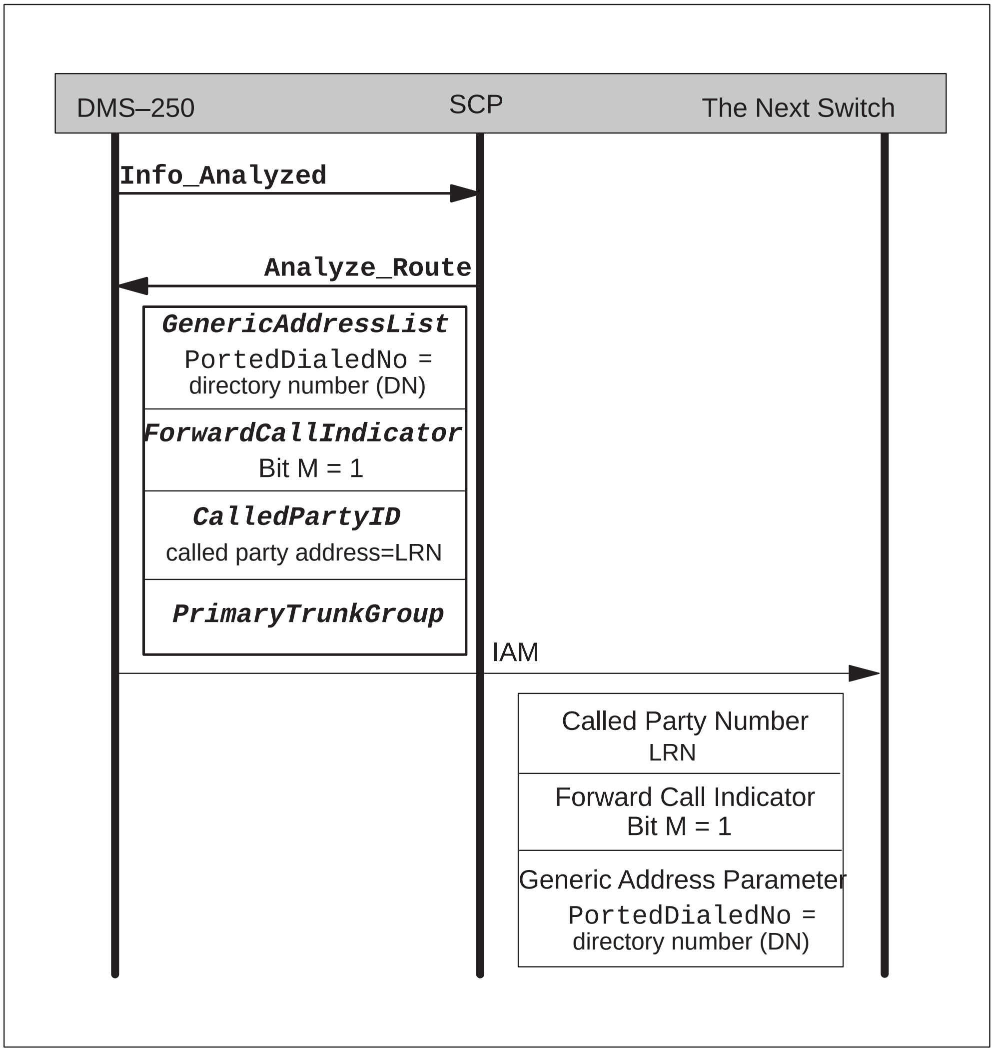
   — places the LRN in the outgoing IAM's Called Party Number (CPN) parameter

   — sets the Forward Call Indicator (FCI) parameter's bit M

   Note: The switch can also use the LRN to route the call using standard routing. See "Standard routing" on page 5-60.

NetworkBuilder queries (continued)

Figure 5-16
GenericAddressList and CalledPartyID returned, ForwardCallIndicator bit M set



NetworkBuilder queries (continued)

OM information

The registers OFCDLOOK, LNPQUERY, and LRNONLNP are pegged.

CDR Summary

Table 5-23 summarizes the CDR population for Scenario 1.

Table 5-23
Scenario 1 CDR summary

LNPCHECK	DIALEDNO	CALLEDNO	OUTPULNO
5	incoming IAM’s CPN	incoming IAM’s CPN	SCP response’s LRN
CN1REQ	**PORTEDNO**		
9	empty		

Scenario 2: NetworkBuilder query returning home LRN

When the switch launches a NetworkBuilder query and the SCP's response contains a home LRN and the ***PrimaryTrunkGroup*** parameter, the switch uses the value in the ***PrimaryTrunkGroup*** parameter to route the call and places the LRN in the outgoing IAM message's Called Party Number (CPN) parameter.

Figure 5-17 illustrates Scenario 2. The following explains, in detail, Scenario 2:

1 The switch receives an incoming IAM that does not have the Forward Call Indicator (FCI) parameter's bit M set. This means the previous switch has not performed an LNP check.

2 The switch determines a NetworkBuilder query must be performed and launches the query.

3 The SCP sends a response message containing an

— LRN in the ***CalledPartyID*** parameter

— ***GenericAddressList*** parameter containing a `PortedDialedNo`

— ***ForwardCallIndicator*** parameter

— ***PrimaryTrunkGroup*** parameter

Note: The response message may contain additional CAIN routing parameters. In this example, the response contains the ***PrimaryTrunkGroup*** parameter.

4 The switch looks up the LRN in table TERMLRN and determines the LRN is one of its home LRNs.

5 The switch uses the value in the ***PrimaryTrunkGroup*** parameter to route the call and

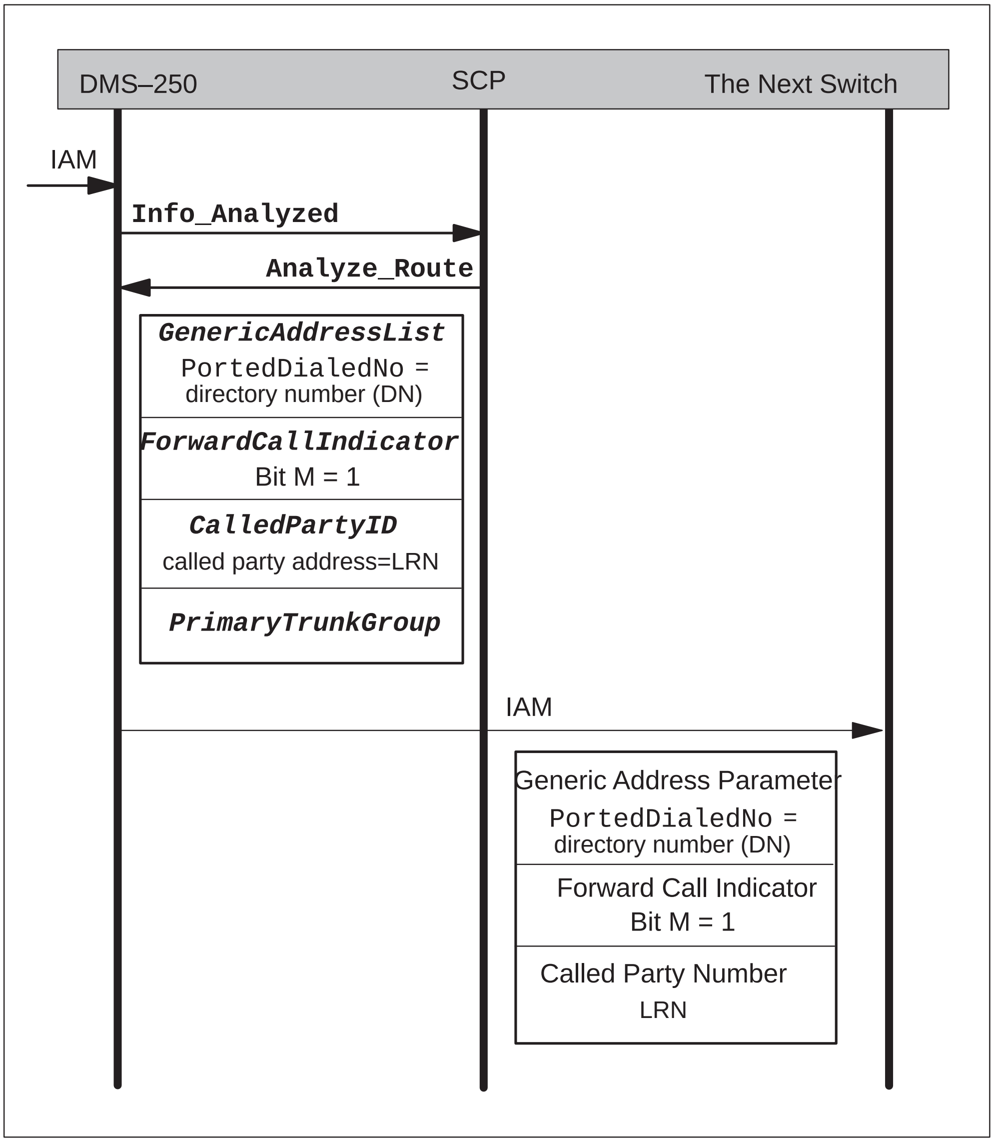
— places the `PortedDialedNo` in the outgoing IAM's Generic Address Parameter (GAP)

— places the LRN in the outgoing IAM's Called Party Number (CPN) parameter

— sets bit M to 1 in the outgoing IAM's Forward Call Indicator (FCI) parameter

NetworkBuilder queries (continued)

Figure 5-17
Scenario 2: NetworkBuilder query returning home LRN



Terminating agent information

If the terminating agent is an SS7 agent, the outgoing IAM contains FCI bit M set, no LNP GAP, and the Called Party Number (CPN) contains the called party address returned in the LNP GAP.

Note: If the call is a direct route, the Called Party Number (CPN) contains the address in the ***`OutpulseNumber`*** parameter.

If the terminating agent is non-SS7 agent, the LNP GAP is outpulsed.

OM information

The registers TERMLRN, OFCDLOOK, LRNONLNP, and LNPQUERY are pegged.

CDR summary

Table 5-24 summarizes the CDR fields captured when a CAIN query returns a Home LRN.

Table 5-24
Scenario 2 CDR summary

DIALEDNO	CALLEDNO	OUTPULNO	TERMLRN
incoming IAM's CPN	SCP response's ***`GenericAddressList`***	SCP response's ***`GenericAddressList`***	SCP response's LRN
LNPCHECK	**ORIGLRN**	**PORTEDNO**	
7	JIP in IAM, if present	empty	

NetworkBuilder queries (continued)

Scenario 3: GenericAddressList and CalledPartyID returned, ForwardCallIndicator with bit M not set

A non-fatal error occurs when the SCP returns an **`Analyze_Route`** message without the ***`ForwardCallIndicator`*** parameter's bit M set, but with

- the ***`CalledPartyID`*** parameter
- the ***`GenericAddressList`*** parameter containing a `PortedDialedNo`

Figure 5-18 illustrates Scenario 3. The following explains, in detail, Scenario 3:

1 The switch launches a NetworkBuilder query.

2 The SCP returns an **`Analyze_Route`** message without the ***`ForwardCallIndicator`*** parameter's bit M set, but with

- — the ***`CalledPartyID`*** parameter
- — the ***`GenericAddressList`*** parameter containing a `PortedDialedNo`
- — the ***`PrimaryTrunkGroup`*** parameter

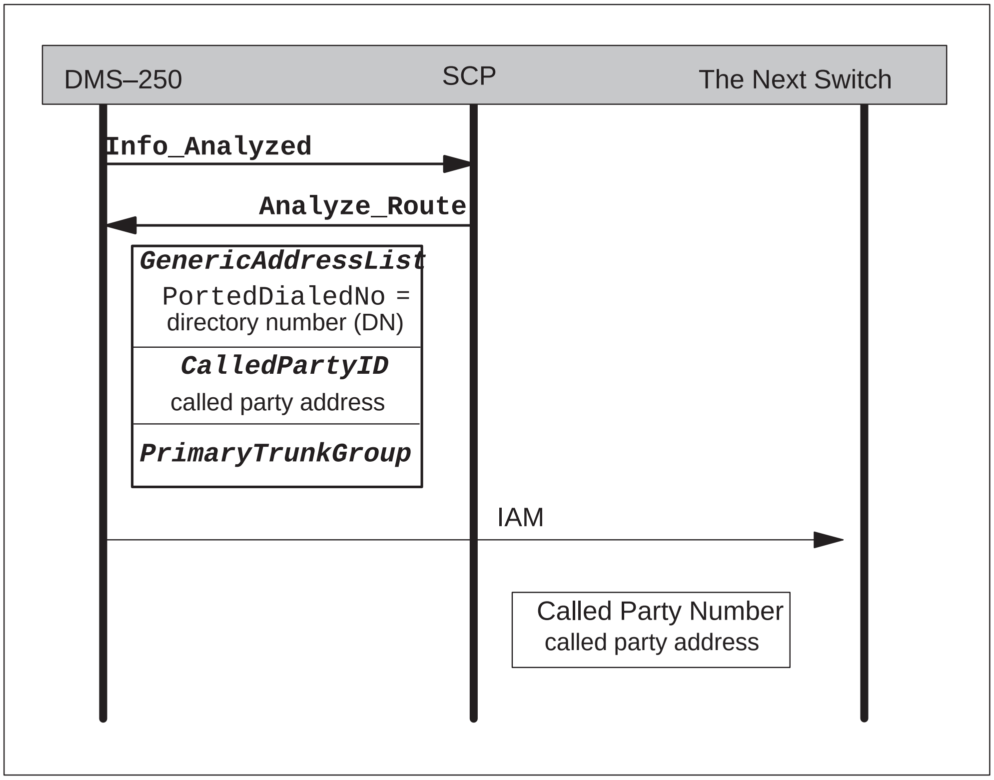
3 The switch determines that the SCP did not return an LRN and considers the information in the ***`CalledPartyID`*** parameter to be the called party address.

4 The switch uses the value in the ***`PrimaryTrunkGroup`*** parameter to route the call and

- — discards the ***`GenericAddressList`*** parameter
- — places the called party address (from the ***`CalledPartyID`*** parameter) in the outgoing IAM's Called Party Number (CPN) parameter
- — does not set bit M in the outgoing IAM's Forward Call Indicator (FCI) parameter

Note: The switch can also use the ***`CalledPartyID`*** parameter to route the call using standard routing. See "Standard routing" on page 5-60.

Figure 5-18
GenericAddressList and CalledPartyID returned, ForwardCallIndicator with bit M not set



NetworkBuilder queries (continued)

OM information

The registers OFCDLOOK, LNPQUERY, and LRNONLNP are pegged.

CDR summary

Table 5-25 summarizes the CDR population for Scenario 3.

Table 5-25
Scenario 3 CDR summary

LNPCHECK	DIALEDNO	CALLEDNO	OUTPULNO
5	incoming IAM's CPN	SCP response's ***CalledPartyID***	SCP response's ***CalledPartyID***
PORTEDNO	**CN1REQ**		
empty	9		

Scenario 4: CalledPartyID returned, ForwardCallIndicator with bit M set, no GenericAddressList returned

The switch determines that the SCP did not return an LRN when the SCP returns an **`Analyze_Route`** message without the ***`GenericAddressList`*** parameter, but with

- the ***`CalledPartyID`*** parameter
- the ***`ForwardCallIndicator`*** parameter's bit M set

Figure 5-19 illustrates Scenario 4. The following explains, in detail, Scenario 4:

1 The switch launches a NetworkBuilder query.

2 The SCP returns an **`Analyze_Route`** message without the ***`GenericAddressList`*** parameter, but with

  — the ***`CalledPartyID`*** parameter

  — the ***`ForwardCallIndicator`*** parameter's bit M set

3 The switch determines that the SCP did not return an LRN and considers the information in the ***`CalledPartyID`*** parameter to be the called party address.

4 The switch uses the value in the ***`PrimaryTrunkGroup`*** parameter to route the call and

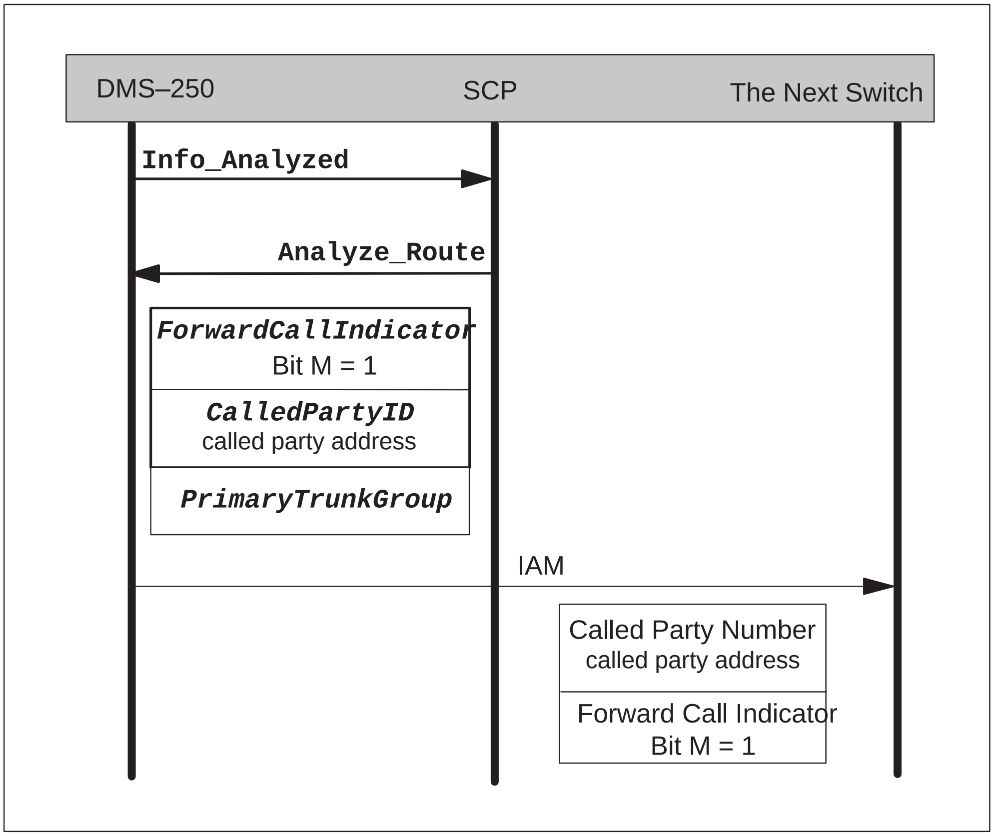
  — places the called party address (from the ***`CalledPartyID`*** parameter) in the outgoing IAM's Called Party Number (CPN) parameter

  — sets bit M of the outgoing IAM's Forward Call Indicator (FCI) parameter

  Note: The switch can also use the ***`CalledPartyID`*** parameter to route the call using standard routing. See "Standard routing" on page 5-60.

NetworkBuilder queries (continued)

Figure 5-19
CalledPartyID returned, ForwardCallIndicator bit M set, no GenericAddressList returned



OM information

The registers are OFCDLOOK, LRNONLNP, and LNPQUERY.

CDR summary

Table 5-26 summarizes the CDR population for Scenario 4.

Table 5-26
Scenario 4 CDR summary

LNPCHECK	CALLEDNO	DIALEDNO	OUTPULNO
5	incoming IAM’s CPN	incoming IAM’s CPN	incoming IAM’s CPN
CN1REQ	**PORTEDNO**		
9	empty		

NetworkBuilder queries (continued)

Scenario 5: CalledPartyID returned, ForwardCallIndicator bit M not set, no GenericAddressList returned

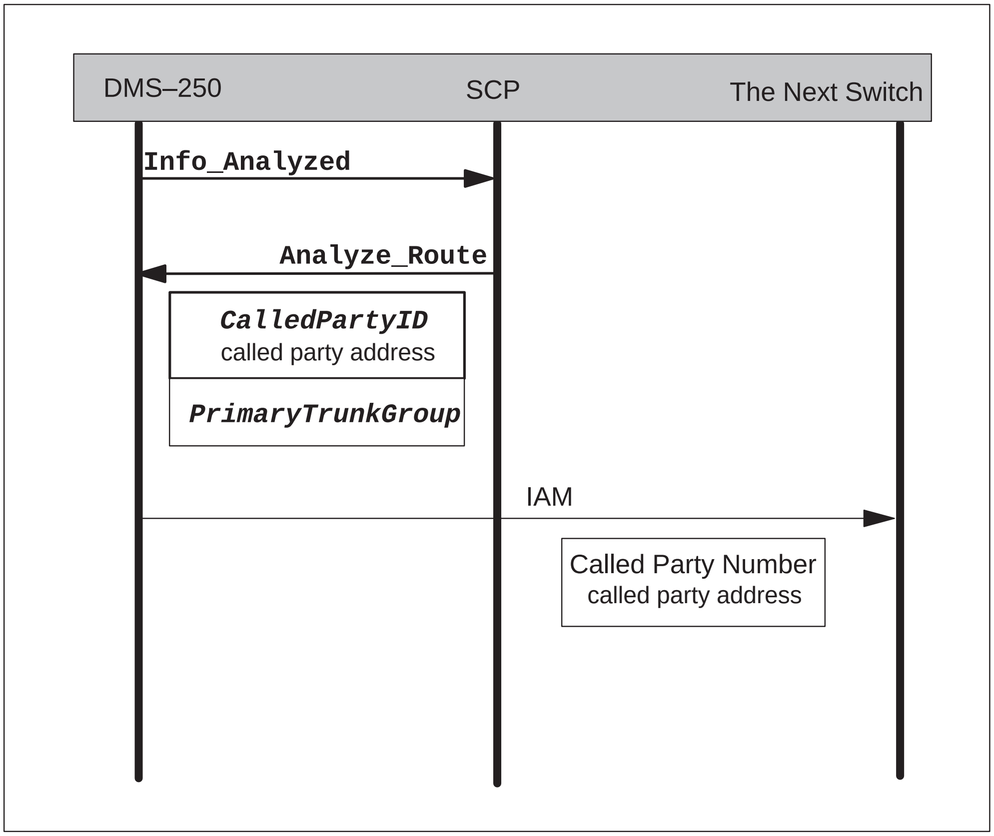
The switch determines that the SCP did not return an LRN when the SCP returns an **`Analyze_Route`** message with the ***`CalledPartyID`*** parameter, but without

- the ***`ForwardCallIndicator`*** parameter's bit M set
- the ***`GenericAddressList`*** parameter

Figure 5-20 illustrates Scenario 5. The following explains, in detail, Scenario 5:

1. The switch launches a NetworkBuilder query.
2. The SCP returns an **`Analyze_Route`** message with the ***`CalledPartyID`*** parameter, but without
   — the ***`ForwardCallIndicator`*** parameter's bit M set
   — the ***`GenericAddressList`*** parameter
3. The switch determines that the SCP did not return an LRN and considers the information in the ***`CalledPartyID`*** parameter to be the called party address.
4. The switch uses the value in the ***`PrimaryTrunkGroup`*** parameter to route the call and places the called party address (from the ***`CalledPartyID`*** parameter) in the outgoing IAM's Called Party Number (CPN) parameter.

Figure 5-20
Scenario 5: CalledPartyID returned, ForwardCallIndicator bit M not set, no GenericAddressList returned



NetworkBuilder queries (continued)

OM information

The registers OFCDLOOK, LRNONLNP, and LNPQUERY are pegged.

CDR summary

Table 5-27 summarizes the CDR population for Scenario 5.

Table 5-27
Scenario 5 CDR summary

LNPCHECK	CALLEDNO	DIALEDNO	OUTPULNO
5	incoming IAM's CPN	incoming IAM's CPN	incoming IAM's CPN
CN1REQ	**PORTEDNO**		
9	empty		

Scenario 6: CalledPartyID not returned, GenericAddressList returned, ForwardCallIndicator bit M set

A non-fatal error occurs when the SCP returns an **`Analyze_Route`** message without the ***`CalledPartyID`*** parameter, but with

- the ***`ForwardCallIndicator`*** parameter's bit M set
- the ***`GenericAddressList`*** parameter containing a `PortedDialedNo`

Figure 5-21 illustrates Scenario 6. The following explains, in detail, Scenario 6:

1 The switch launches a NetworkBuilder query.

2 The SCP returns an **`Analyze_Route`** message without the ***`CalledPartyID`*** parameter, but with

— the ***`ForwardCallIndicator`*** parameter's bit M set

— the ***`GenericAddressList`*** parameter containing a `PortedDialedNo`

3 The switch determines that the SCP did not return an LRN.

4 The switch uses the value in the ***`PrimaryTrunkGroup`*** parameter to route the call and

— does not send the Generic Address Parameter (GAP) in the outgoing IAM

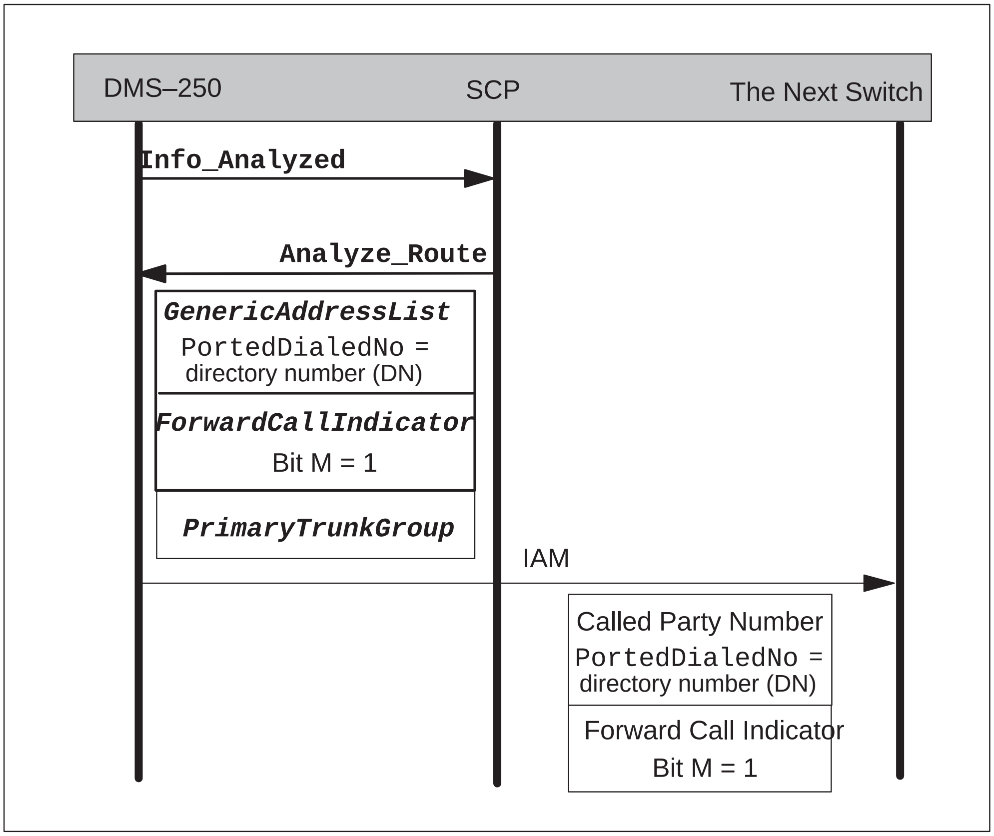
— sets bit M in the in the outgoing IAM's Forward Call Indicator (FCI) parameter

— places the `PortedDialedNo` in the outgoing IAM's Called Party Number (CPN) parameter

Note: The switch can also use a standard route other than the ***`CalledPartyID`*** parameter or the called party address to route the call using standard routing. See "Standard routing" on page 5-60.

NetworkBuilder queries (continued)

Figure 5-21
Scenario 6: CalledPartyID not returned, GenericAddressList returned, ForwardCallIndicator bit M set



OM information

The registers OFCDLOOK, LNPQUERY, and LRNONLNP are pegged.

CDR summary

Table 5-28 summarizes the CDR population for Scenario 6.

Table 5-28
Scenario 6 CDR summary

LNPCHECK	CALLEDNO	DIALEDNO	OUTPULNO
5	incoming IAM's CPN	incoming IAM's CPN	incoming IAM's CPN
CN1REQ	**PORTEDNO**		
9	empty		

NetworkBuilder queries (continued)

Scenario 7: CalledPartyID not returned, GenericAddressList returned, ForwardCallIndicator bit M not set

A non-fatal error occurs when an **Analyze_Route** message returns with the ***GenericAddressList*** parameter containing a `PortedDialedNo`, but without

- the ***CalledPartyID*** parameter
- the ***ForwardCallIndicator*** parameter's bit M set

Figure 5-22 Scenario 7. The following explains, in detail, Scenario 7:

1 The switch launches a NetworkBuilder query.

2 The SCP returns an **Analyze_Route** message with the ***GenericAddressList*** parameter containing a `PortedDialedNo`, but without

— the ***CalledPartyID*** parameter

— the ***ForwardCallIndicator*** parameter's bit M set

3 The switch determines that the SCP did not return an LRN.

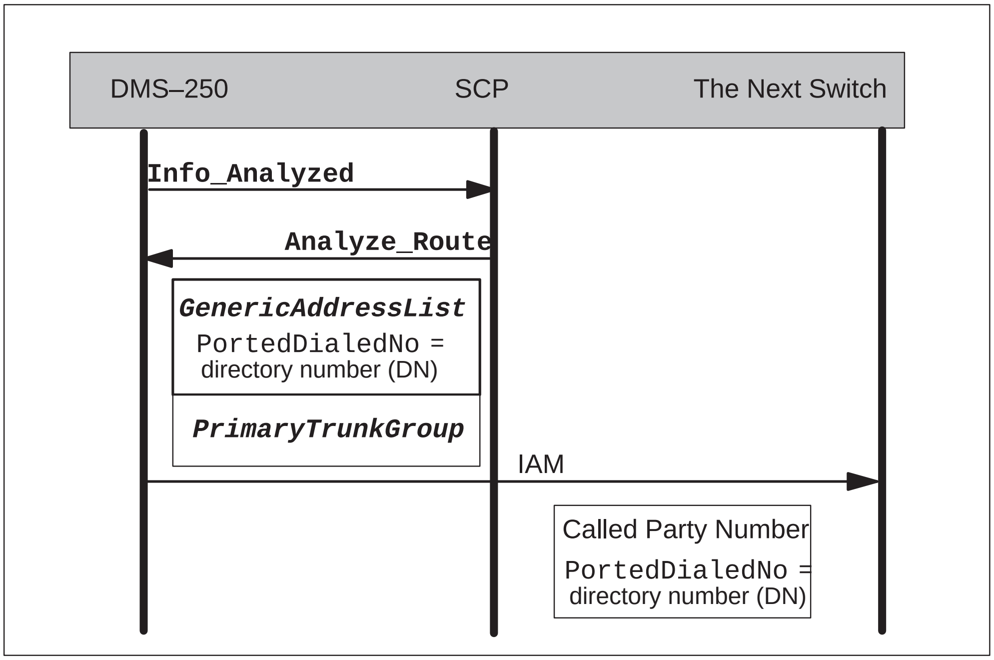
4 The switch uses the value in the ***PrimaryTrunkGroup*** parameter to route the call and

— discards the ***GenericAddressList*** parameter

— places the `PortedDialedNo` in the outgoing IAM's Called Party Number (CPN) parameter

Note: The switch can also use a standard route other than the ***CalledPartyID*** parameter or the called party address to route the call. See "Standard routing" on page 5-60.

Figure 5-22
Scenario 7: CalledPartyID not returned, GenericAddressList returned, ForwardCallIndicator bit M not set



NetworkBuilder queries (continued)

OM information

The registers OFCDLOOK, LRNONLNP, and LNPQUERY are pegged.

CDR summary

Table 5-29 summarizes the CDR population for Scenario 7.

Table 5-29
Scenario 7 CDR summary

LNPCHECK	CALLEDNO	DIALEDNO	OUTPULNO
5	incoming IAM’s CPN	incoming IAM’s CPN	incoming IAM’s CPN
CN1REQ	**PORTEDNO**		
9	empty		

Scenario 8: CalledPartyID and GenericAddressList not returned, ForwardCallIndicator bit M set

The switch determines that the SCP did not return an LRN when the SCP returns an **Analyze_Route** message with the ***ForwardCallIndicator*** parameter's bit M set, but without

- the ***CalledPartyID*** parameter
- the ***GenericAddressList*** parameter

Figure 5-23 illustrates Scenario 8. The following explains, in detail, Scenario 8:

1 The switch launches a NetworkBuilder query.

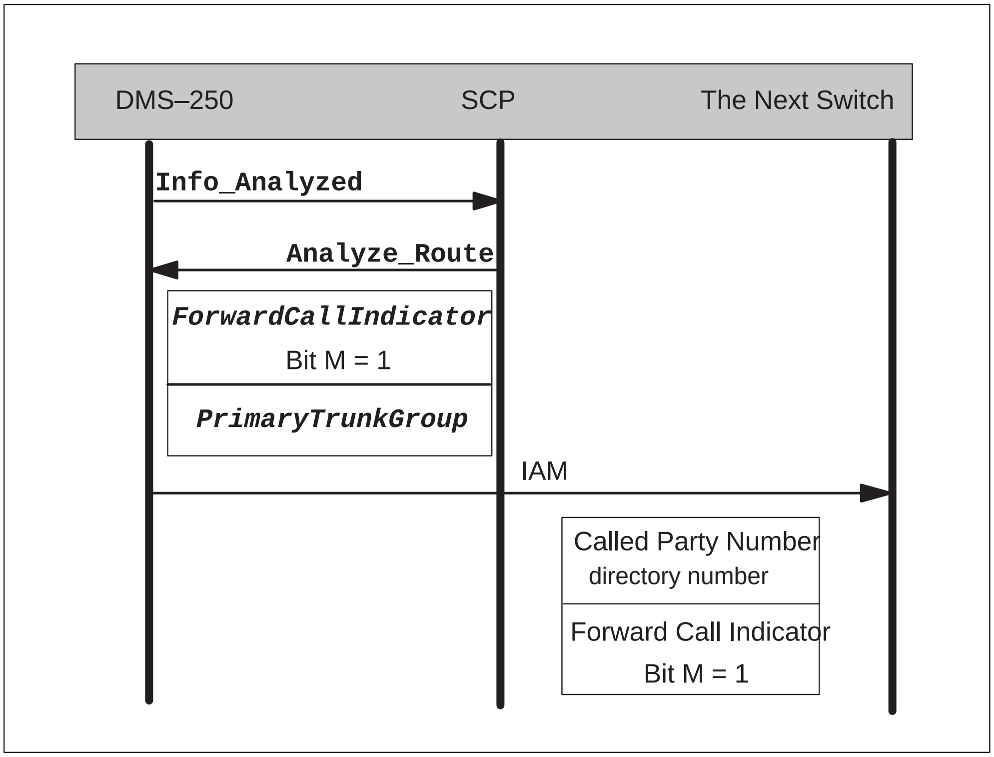
2 The SCP returns an **Analyze_Route** message with the ***ForwardCallIndicator*** parameter's bit M set, but without
   — the ***CalledPartyID*** parameter
   — the ***GenericAddressList*** parameter

3 The switch determines that the SCP did not return an LRN.

4 The switch uses the value in the ***PrimaryTrunkGroup*** parameter to route the call and
   — sets bit M in the outgoing IAM's Forward Call Indicator (FCI) parameter
   — places the directory number (DN) in the outgoing IAM's Called Party Number (CPN) parameter

NetworkBuilder queries (continued)

Figure 5-23
Scenario 8: CalledPartyID and GenericAddressList not returned, ForwardCallIndicator bit M set



OM information

The registers OFCDLOOK, LRNONLNP, and LNPQUERY are pegged.

CDR summary

Table 5-30 summarizes the CDR population for Scenario 8.

Table 5-30
Scenario 8 CDR summary

LNPCHECK	CALLEDNO	DIALEDNO	OUTPULNO
5	incoming IAM's CPN	incoming IAM's CPN	incoming IAM's CPN
CN1REQ	**PORTEDNO**		
9	empty		

NetworkBuilder queries (continued)

Scenario 9: CalledPartyID and GenericAddressList not returned, ForwardCallIndicator bit M not set

When the SCP returns an **`Analyze_Route`** message with only direct termination routing information, the switch routes the call without launching an LNP query.

Figure 5-24 illustrates Scenario 9. The following explains, in detail, Scenario 9:

1 The switch launches a NetworkBuilder query.

2 The SCP returns an **`Analyze_Route`** message containing only the ***`PrimaryTrunkGroup`*** parameter. The **`Analyze_Route`** message does not contain

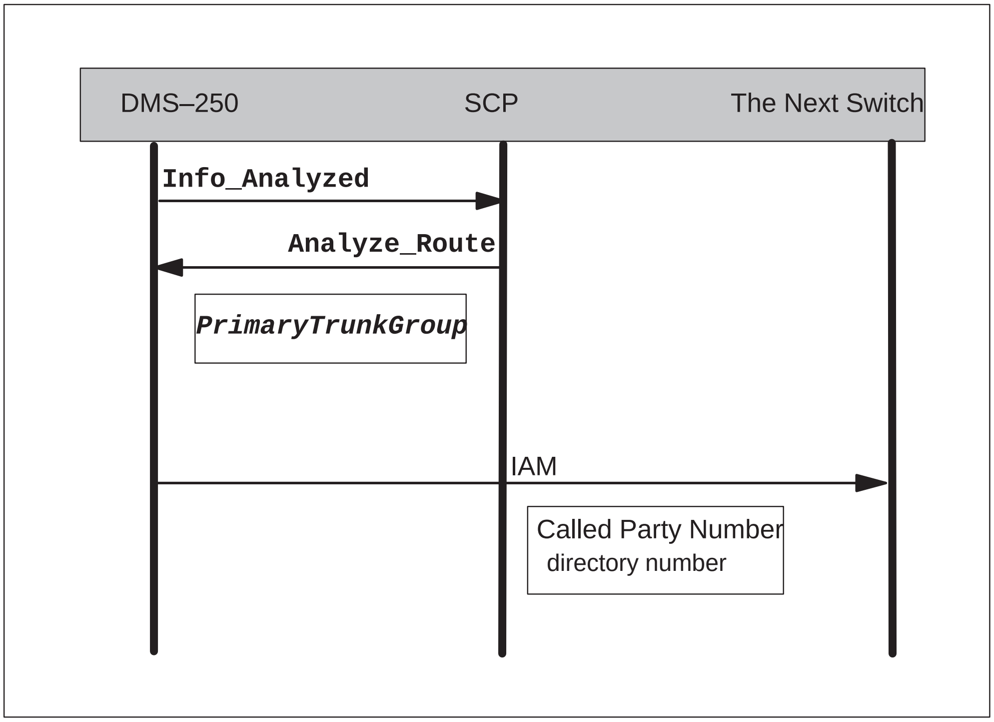
- the ***`ForwardCallIndicator`*** parameter's bit M set
- the ***`CalledPartyID`*** parameter
- the ***`GenericAddressList`*** parameter

3 The switch uses the value in the ***`PrimaryTrunkGroup`*** parameter to route the call and

- does not set bit M in the outgoing IAM's Forward Call Indicator (FCI) parameter
- does not send the Generic Address Parameter (GAP) in the outgoing IAM
- sends the directory number in the outgoing IAM's Called Party Number (CPN) parameter

NetworkBuilder queries (continued)

Figure 5-24
Scenario: CalledPartyID and GenericAddressList not returned, ForwardCallIndicator bit M not set



NetworkBuilder queries (continued)

OM information

The registers OFCDLOOK, LRNONLNP, and LNPCHECK are pegged.

CDR summary

Table 5-31 summarizes the CDR population for Scenario 9.

Table 5-31
Scenario 9 CDR summary

LNPCHECK	DIALEDNO	CALLEDNO	OUTPULNO
5	incoming IAM's CPN	incoming IAM's CPN	incoming IAM's CPN
CN1REQ	**PORTEDNO**		
9	empty		

Scenario 10: Direct termination and standard routing

When the SCP returns both direct termination and standard routing information an **`Analyze_Route`** message, the switch first attempts direct routes. If the direct routes are unsuccessful, such as in the busy scenarios, the ***`CalledPartyID`*** and `OverflowRoutingNo` are evaluated at the *Office_Code* trigger.

Note: For both direct termination and standard routing scenarios, reevaluation at the *Office_Code* trigger will not occur if the SCP returns an LRN or an ***`ForwardCallIndicator`*** parameter with bit M set.

Figure 5-25 illustrates Scenario 10. The following explains, in detail, Scenario 10:

1 The switch launches a NetworkBuilder query.

2 The SCP returns an **`Analyze_Route`** message with

— the ***`CalledPartyID`*** parameter

— the ***`ForwardCallIndicator`*** parameter with bit M set

— the ***`GenericAddressList`*** parameter containing a `PortedDialedNo`

— the ***`PrimaryTrunkGroup`*** parameter

3 The switch determines that the value in the ***`CalledPartyID`*** parameter is an LRN.

4 The switch attempts to use direct termination routing (the ***`PrimaryTrunkGroup`*** parameter) to route the call and

— places the `PortedDialedNo` in the Generic Address Parameter (GAP) of the outgoing IAM

— places the LRN in the Called Party Number (CPN) parameter of the outgoing IAM

— sets bit M in the Forward Call Indicator (FCI) parameter of the outgoing IAM

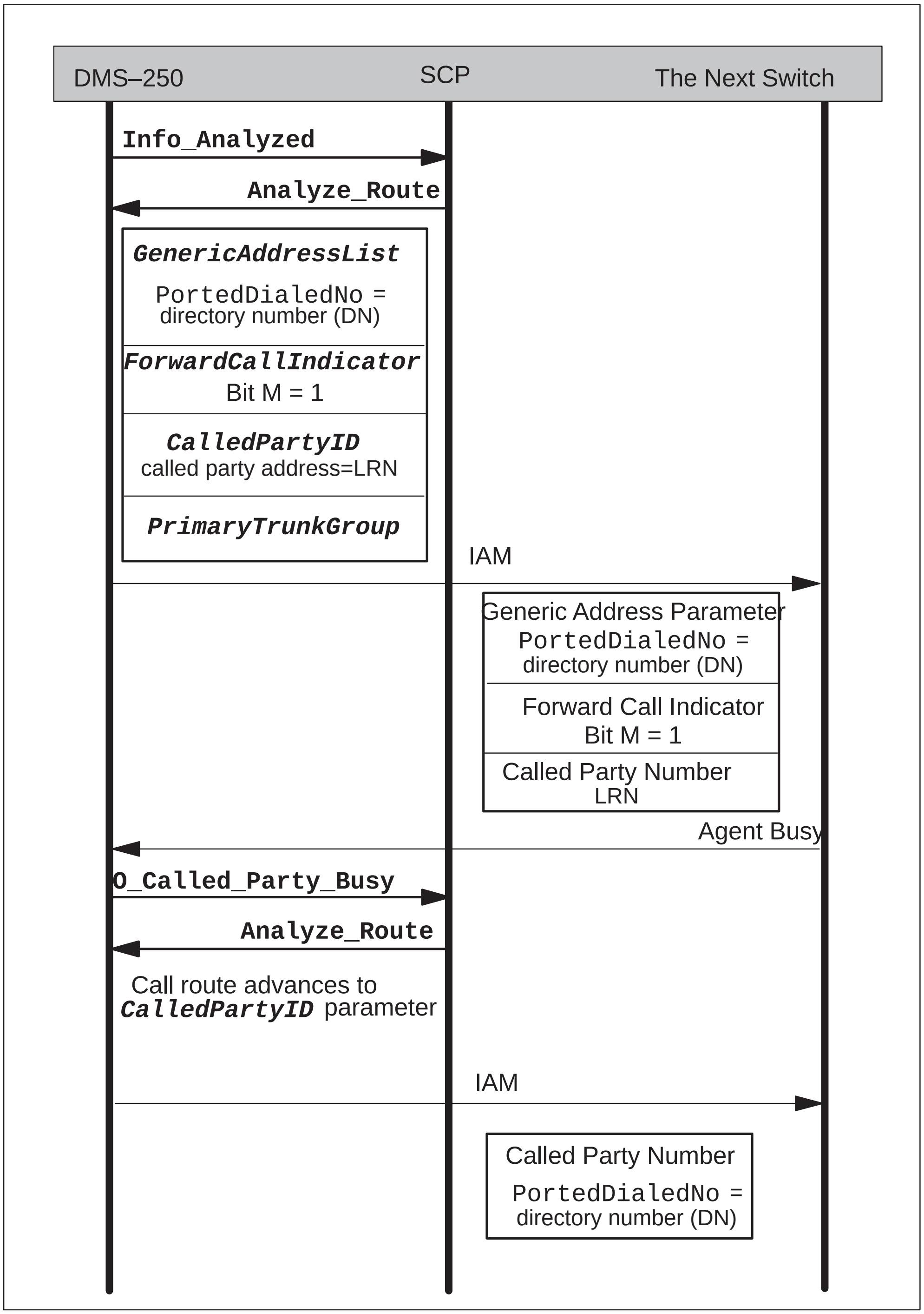
5 The switch receives an agent busy message and sends a second NetworkBuilder query to the SCP.

6 The SCP returns an **`Analyze_Route`** message and the call route advances to the ***`CalledPartyID`*** parameter.

7 The switch uses standard routing to route the call and sends the `PortedDialedNo` in the Called Party Number (CPN) parameter of the outgoing IAM.

NetworkBuilder queries (continued)

Figure 5-25
Scenario 10: Direct termination and standard routing



OM information

The registers OFCDLOOK, LNPQUERY, and LRNONLNP are pegged.

CDR summary

Table 5-32 summarizes the CDR population for Scenario 10.

Table 5-32
Scenario 10 CDR summary

LNPCHECK	DIALEDNO	CALLEDNO	OUTPULNO
5	incoming IAM's CPN	SCP response's (after **O_Called_Party_Busy**) ***CalledPartyID***	SCP response's (after **O_Called_Party_Busy**) ***CalledPartyID***
CN1REQ	**PORTEDNO**		
empty	empty		

NetworkBuilder queries (continued)

Scenario 11: IN/1 Start message interacting with LNP query

When a ***Digits*** parameter containing `Carrier` digits is included in a **Connect** response message that has a value indicating the selected carrier is a LEC (indicated by a value of “0110” or “110”), call processing resumes at the **Analyze_Information** PIC. This allows for reevaluation of the *Office_Code* trigger before the call is routed.

Figure 5-26 illustrates Scenario 11. The following explains, in detail, Scenario 11:

1 The switch launches a **Tollfree_Service** TDP-Request to the SCP. The following parameters are contained within the request:

— ***ServiceKey*** parameter (dialed 8yy number requiring number translation)

— ***Digits*** parameter containing `LATA` digits (associated with the originating trunk)

— ***Digits*** parameter containing `CallingPartyNumber` (ANI)

— ***OriginatingStationType*** (II digits)

2 The SCP responds with the following information in a **Connect** message:

— ***Digits*** parameter containing `RoutingNumber` (POTS DN)

— ***Digits*** parameter containing `Carrier` digits (index to table CICROUTE) of “0110”

3 The switch reevaluates the **Info_Analyzed** TDP and triggers at *Office_Code*. The switch launches an LNP query to the LNP SCP.

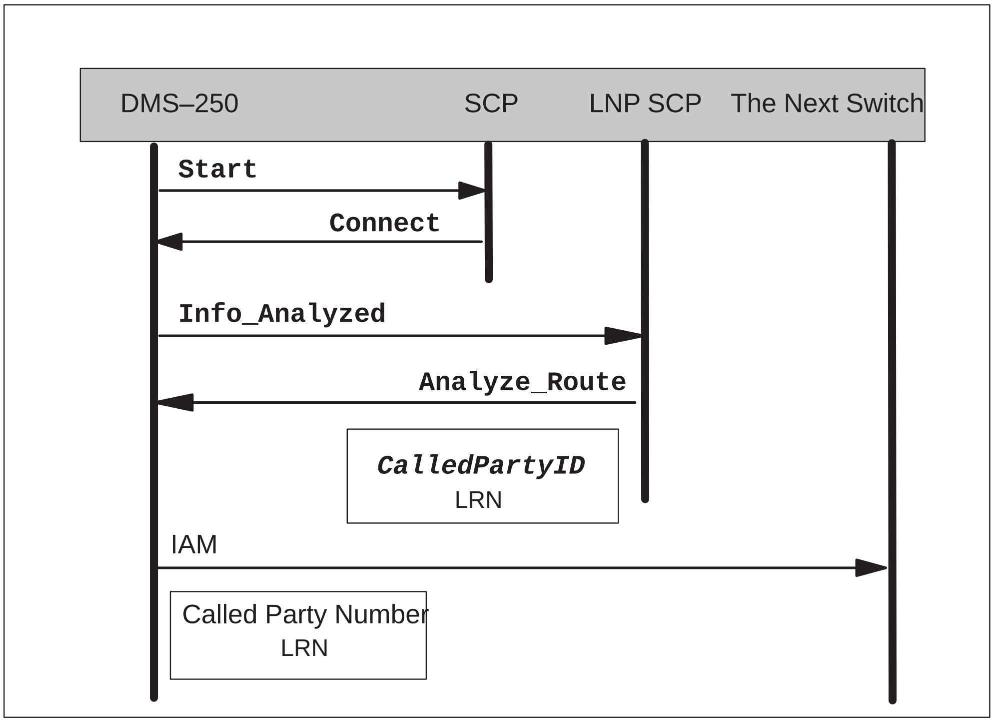
4 The LNP SCP responds with an **Analyze_Route** message containing an LRN in the ***CalledPartyID*** parameter.

5 The switch routes the call to the next switch.

Note: For more information on IN/1 queries, see the *UCS DMS-250 NetworkBuilder Application Guide*.

NetworkBuilder queries (continued)

Figure 5-26
Scenario 11: IN/1 Start message interacting with LNP query



NetworkBuilder queries (end)

OM information

The registers OFCDLOOK, LNPQUERY, and LRNONLNP are pegged.

CDR summary

Table 5-33 summarizes the CDR population for Scenario 11.

Table 5-33
Scenario 11 CDR summary

LNPCHECK	DIALEDNO	CALLEDNO	OUTPULNO
5	incoming IAM's CPN	SCP response's ***CalledPartyID***	SCP response's ***CalledPartyID***
CN1REQ	**PORTEDNO**		
9	empty		

Incoming LNP information

This section describes scenarios in which LNP information is incoming from a previous switch. The section contains

- an overview of incoming LNP information
- figures illustrating scenarios in which LNP information is incoming from a previous switch

Overview of incoming LNP information

Incoming LNP information is LNP information received in the SS7 initial address message (IAM) from a previous switch. When the incoming SS7 IAM contains LNP information the IAM's Forward Call Indicator (FCI) parameter's bit M is set. The FCI with bit M set informs the switch that the previous switch performed an LNP check.

When the incoming IAM message contains LNP information, the previous switch either

- determined the directory number (DN) is a ported number and is sending an LRN

  or

- determined the DN is not a ported number

When the previous switch determined the DN is a ported number and sends an LRN, the incoming IAM message contains

- the Forward Call Indicator (FCI) parameter with bit M set
- the Generic Address Parameter (GAP) containing a `PortedDialedNo` (the directory number)
- the Called Party Number (CPN) containing the LRN

When the previous switch determined the DN is not a ported number, the incoming IAM message contains

- the FCI parameter with bit M set
- the DN

Incoming LNP information (continued)

NetworkBuilder queries after receiving LNP information in the incoming IAM message

If it is determined that a NetworkBuilder query is needed after the switch has received LNP information in the incoming IAM message, then the switch notifies the SCP, through the `lnpReceived` extension parameter in the query message, that the switch has received LNP information.

Note: The `lnpReceived` extension parameter is a boolean field supported in NetworkBuilder query messages **`O_Feature_Requested`**, **`Info_Collected`**, and **`Info_Analyzed`**. The `lnpReceived` extension parameter requires the CAIN_PROTOCOL_STREAM of UCS07 or higher.

If an LRN is present in the incoming IAM, it is not used for trigger evaluation nor is it used as the ***`CalledPartyID`*** for NetworkBuilder queries. The Called Party Address from the IAM LNP GAP is used.

Response messages are handled a little differently when an LNP check has been performed on a previous switch. For example, the following messages behave the same as in normal NetworkBuilder queries with the exception of LNP information being kept rather than discarded:

- **`Continue`**
- **`Disconnect`**
- **`Send_To_Resource`**
- Error messages

Following is a list of differences for the **`Analyze_Route`** message when an LNP check has been performed on a previous switch:

- The original LNP information is discarded. In other words, it is not used by NetworkBuilder for further processing.
- The CALLEDNO and PORTEDNO CDR fields are overwritten when a new LRN is returned with the **`Analyze_Route`** message.
- The PORTEDNO CDR field is empty when a new LRN is not returned.

When *Network_Busy*, *O_Called_Party_Busy*, or *O_No_Answer* triggers are encountered, LNP processing occurs as described, except that the original LNP information will be discarded.

Incoming LNP information scenarios

This section contains seven scenarios:

- Scenario 1: Processing of incoming LNP information without an LRN
- Scenario 2: Tandeming of an incoming LRN
- Scenario 3: Tandeming of incoming LNP information without an LRN
- Scenario 4: Processing of an incoming LRN
- Scenario 5: Processing of an incoming home LRN
- Scenario 6: Processing of incoming LRN followed by a NetworkBuilder query
- Scenario 7: Processing of an incoming home LRN followed by a NetworkBuilder query

Each scenario contains

- a brief description of the scenario
- step-by-step details of the scenario
- figures illustrating the scenario
- call detail report (CDR) summary

Note: Some scenarios contain operational measurements (OM) and terminating agent information.

For clarity, the scenarios in this chapter use the exact SCP **`Analyze_Route`** message's parameters' names and SS7 IAM message's parameters' names, rather than the abbreviated names used throughout this document. Table 5-34 maps the abbreviation to the parameters' names.

Table 5-34
Mapping of abbreviations to parameters' names

Abbreviation	SCP response parameter name	SS7 IAM parameter name
LNP GAP	***`GenericAddressList`*** parameter containing a `PortedDialedNo`	Generic Address Parameter (GAP)
FCI	***`ForwardCallIndicator`***	Forward Call Indicator (FCI)

Incoming LNP information (continued)

Scenario 1: processing of incoming LNP information without an LRN

When the previous switch has determined the DN is not a ported number, the incoming IAM message contains

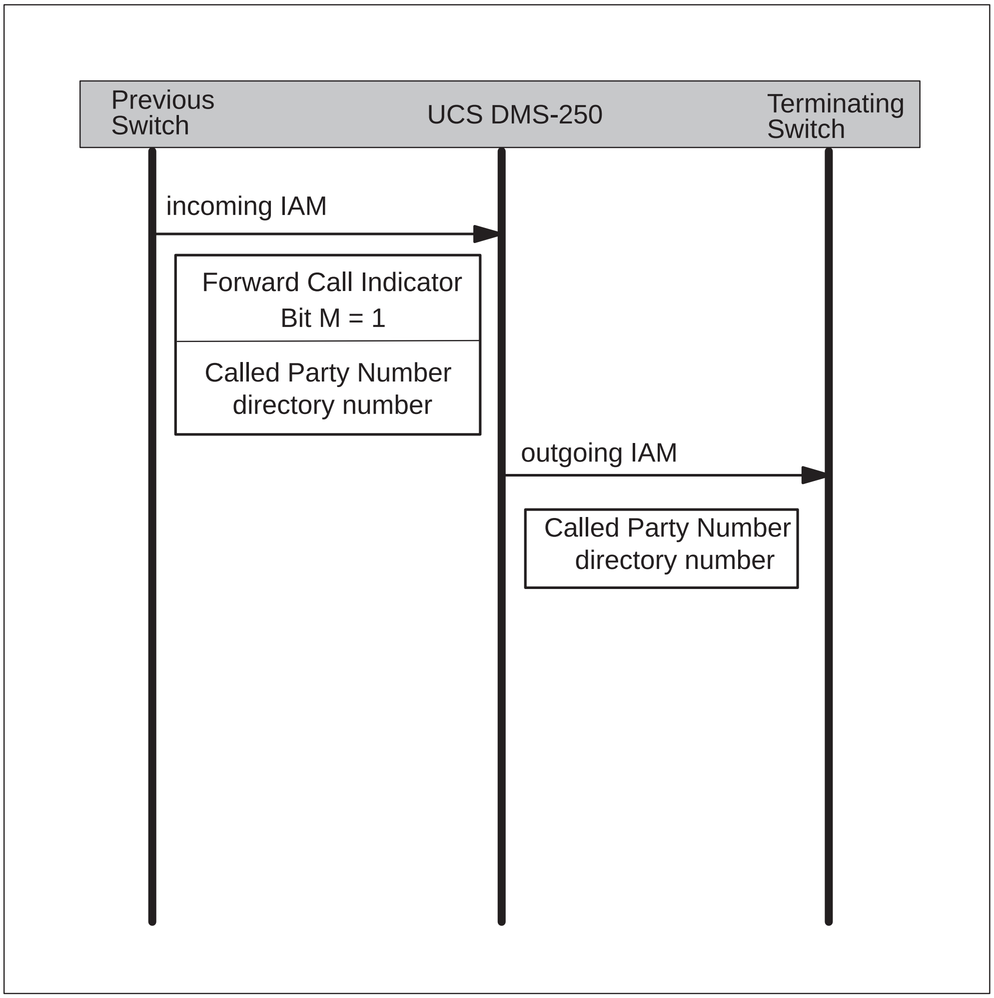
- the FCI parameter with bit M set
- the DN in the Called Party Number (CPN) parameter

Figure 5-27 illustrates Scenario 1. The following explains, in detail, Scenario 1:

1. The switch receives LNP information in the incoming IAM message from the previous switch. The IAM message contains
   - the Forward Call Indicator (FCI) parameter with bit M set
   - the DN in the Called Party Number (CPN) parameter
2. The switch examines the contents of the incoming IAM message. Since the Forward Call Indicator (FCI) parameter's bit M is set, the switch knows the previous switch performed an LNP check.
3. The switch uses standard routing to route the call to the terminating switch.

Incoming LNP information (continued)

Figure 5-27
Scenario 1: Incoming LNP information without an LRN



Incoming LNP information (continued)

OM information

The register INLNPINF is pegged.

CDR summary

Table 5-35 summarizes the CDR population for Scenario 1.

Table 5-35
Scenario 1 CDR summary

LNPCHECK	DIALEDNO	CALLEDNO	OUTPULNO	TERMLRN	PORTEDNO
1	incoming IAM’s CPN	incoming IAM’s CPN	incoming IAM’s CPN	empty	empty

Scenario2: Tandeming of an incoming LRN

When the switch receives an incoming LRN, the switch can tandem the LRN to the next switch.

Figure 5-28 illustrates Scenario 2. The following explains, in detail, Scenario 2:

1 The switch receives an LRN in the incoming IAM message from the previous switch. The IAM message contains

— the Forward Call Indicator (FCI) parameter with bit M set

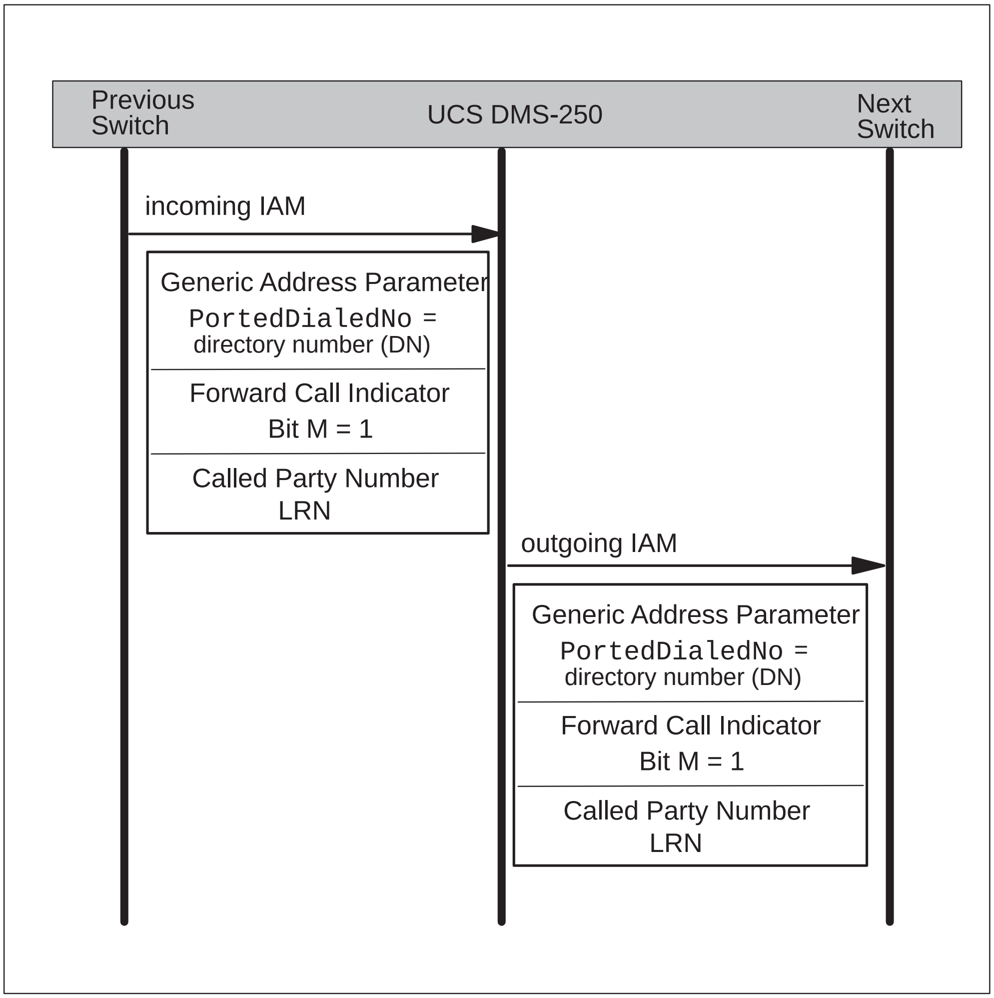
— the Generic Address Parameter (GAP) containing a `PortedDialedNo`

— the Called Party Number (CPN) parameter containing the LRN

2 The switch tandems the information to the next switch.

Incoming LNP information (continued)

Figure 5-28
Scenario 2: Tandeming of an incoming LRN



Incoming LNP information (continued)

OM information

The register INLNPINF is pegged.

CDR summary

Table 5-36 summarizes the CDR population for Scenario 2.

Table 5-36
Scenario 2 CDR summary

DIALEDNO field	CALLEDNO field	OUTPULNO field	PORTEDNO field
incoming IAM’s LRN	incoming IAM’s LRN	incoming IAM’s LRN	incoming IAM’s GAP

Incoming LNP information (continued)

Scenario 3: Tandeming of LNP information without an LRN

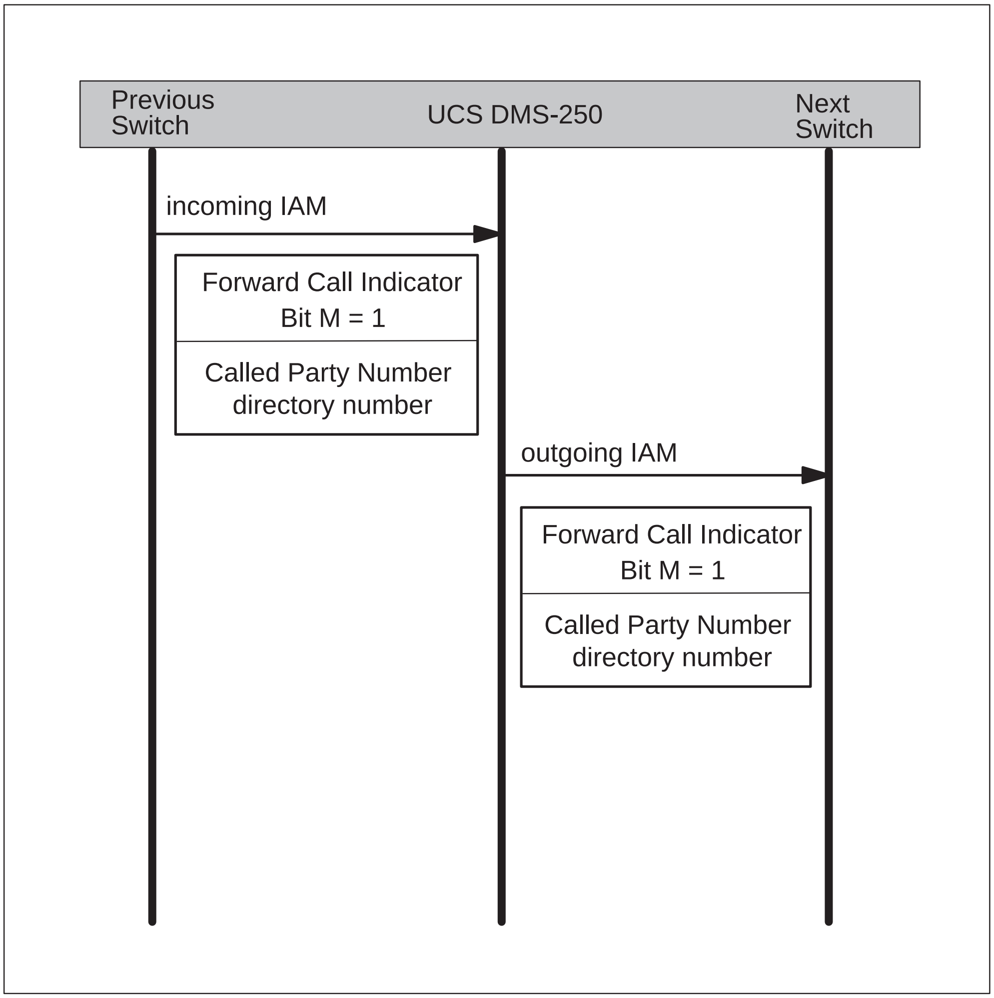
When the switch receives LNP information, but no LRN in the incoming IAM message, the switch can tandem the IAM message to the next switch.

Figure 5-29 illustrates Scenario 3. The following explains, in detail, Scenario 3:

1. The switch receives LNP information in the incoming IAM message from the previous switch. The IAM message contains
   - the Forward Call Indicator (FCI) parameter with bit M set
   - the DN in the Called Party Number (CPN) parameter
2. The switch tandems the information to the next switch.

Incoming LNP information (continued)

Figure 5-29
Scenario 3: Tandeming of incoming LNP information without an LRN



Incoming LNP information (continued)

OM information

The register INLNPINF is pegged.

CDR summary

Table 5-37 summarizes the CDR population for Scenario 3.

Table 5-37
Scenario 3 CDR summary

LNPCHECK	DIALEDNO	CALLEDNO	OUTPULNO	TERMLRN	PORTEDNO
1	incoming IAM's CPN	incoming IAM's CPN	incoming IAM's CPN	empty	empty

Scenario 4: Processing of an incoming LRN

When the incoming IAM message contains an LRN, the call routes based on the LRN.

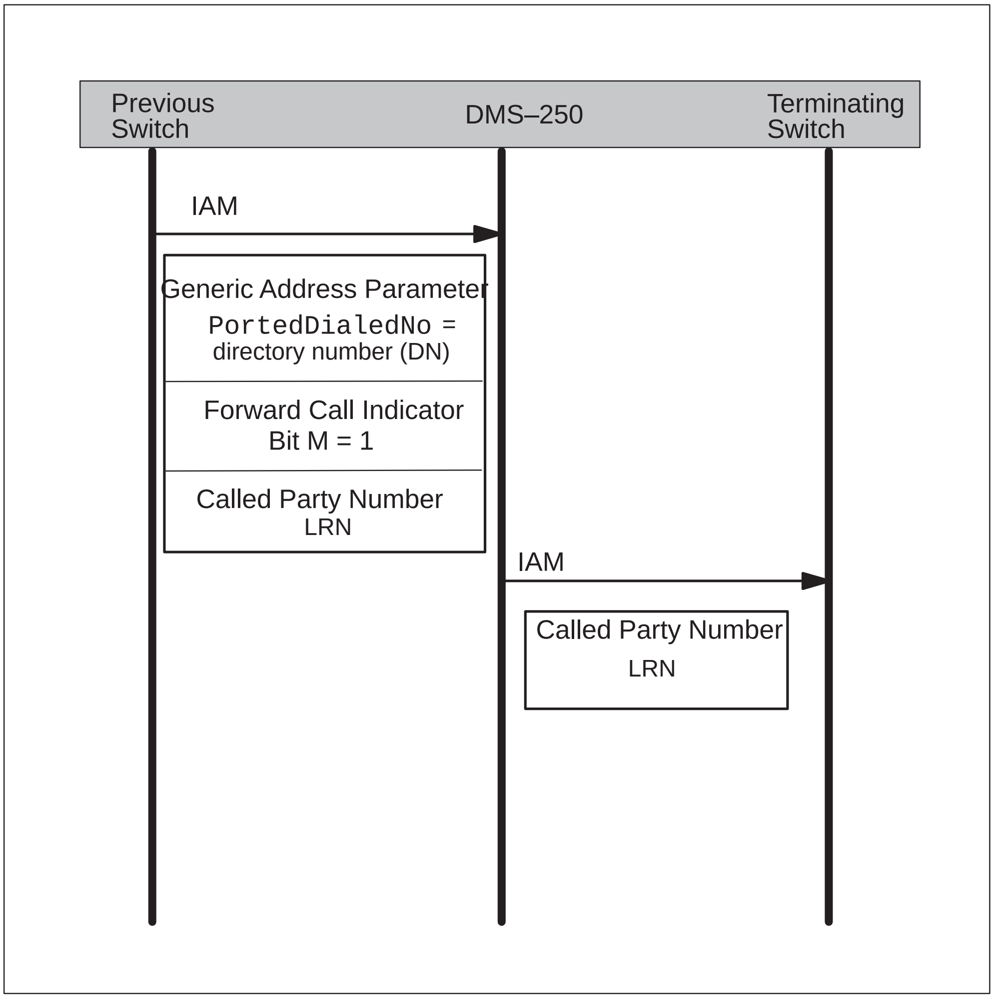
Figure 5-30 illustrates Scenario 3. The following explains, in detail, Scenario 3:

1 The switch receives an LRN in the incoming IAM message. The IAM contains
   - — the Forward Call Indicator (FCI) parameter with bit M set
   - — the Generic Address Parameter (GAP) containing a `PortedDialedNo`
   - — the Called Party Number (CPN) parameter containing the LRN

2 The switch routes uses the LRN to route the call and places the LRN in the Called Party Number (CPN) parameter of the outgoing IAM message.

Incoming LNP information (continued)

Figure 5-30
Scenario 4: Processing of an incoming LRN



Incoming LNP information (continued)

OM information

The registers INLNPINF and TERMLRN are pegged.

CDR summary

Table 5-38 summarizes the CDR population for Scenario 4.

Table 5-38
Scenario 4 CDR summary

LNPCHECK	DIALEDNO	CALLEDNO	OUTPULNO
6	incoming IAM's CPN	incoming IAM's GAP	incoming IAM's GAP
TERMLRN	**PORTEDNO**		
incoming IAM's CPN	empty		

Incoming LNP information (continued)

Scenario 5: Processing of an incoming home LRN

When an incoming IAM contains the switch's home LRN, the switch uses the `PortedDialedNo` (the directory number) to route the call.

Note: For information on the processing of an incoming home LRN followed by a NetworkBuilder query, see Scenario 7 on page 5-116.

Figure 5-31 illustrates Scenario 6. The following explains, in detail, Scenario 6:

3 The switch receives an LRN in the incoming IAM message. The IAM message contains

— the Forward Call Indicator (FCI) parameter with bit M set

— the Called Party Number (CPN) parameter containing the LRN

— a Generic Address Parameter (GAP) containing the `PortedDialedNo`

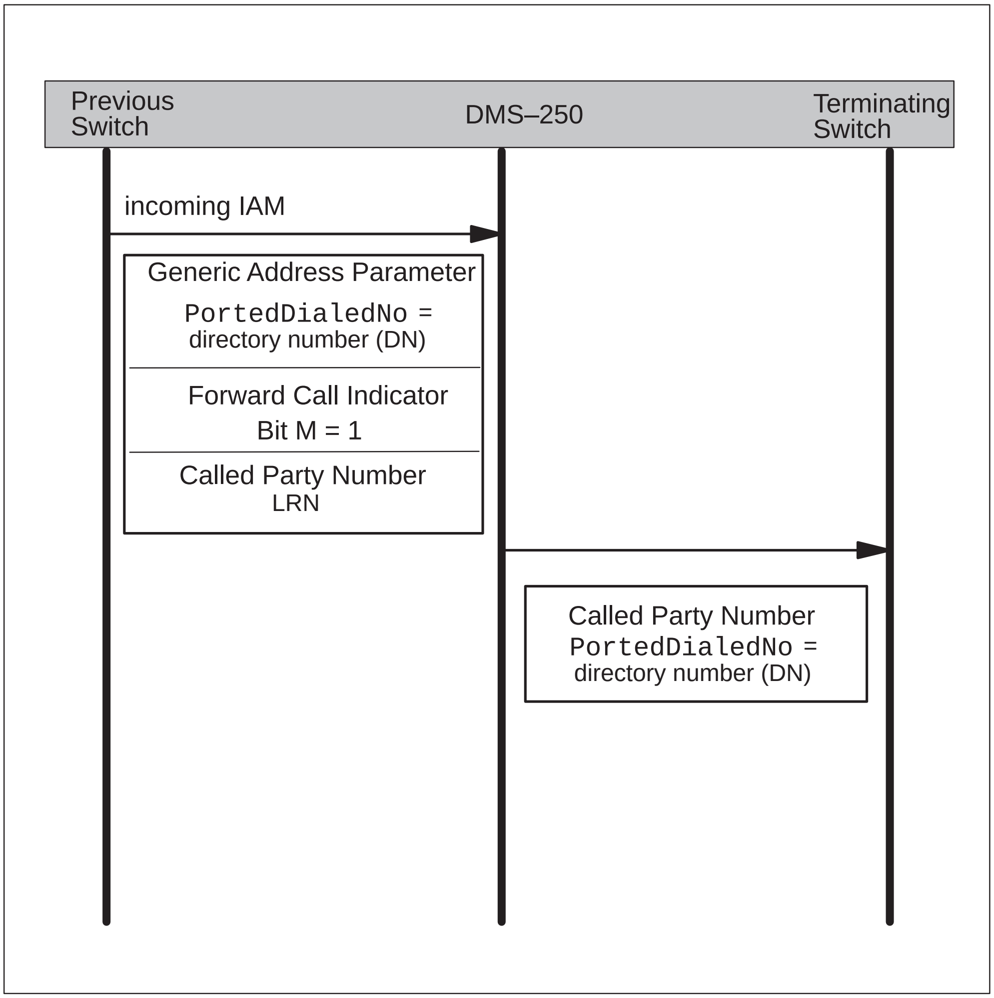
4 The switch determines the LRN is a home LRN.

Note: To determine whether the LRN is a home LRN, the switch verifies the LRN is entered in table TERMLRN.

5 The switch uses the `PortedDialedNo` to route the call.

Incoming LNP information (continued)

Figure 5-31
Scenario 5: Processing of an incoming home LRN



Incoming LNP information (continued)

Terminating agent information

If the terminating agent is an SS7 agent, the outgoing IAM contains Forward Call Indicator (FCI) Bit M set and the Called Party Number (CPN) parameter contains the `PortedDialedNo` from the incoming IAM message’s Generic Address Parameter (GAP).

If the terminating agent is a non-SS7 agent, the `PortedDialedNo` from the incoming IAM message’s Generic Address Parameter (GAP) is outpulsed to the terminating switch.

OM information

The registers TERMLRN and INLNPINF are pegged.

CDR summary

Table 5-39 summarizes the CDR population for Scenario 5.

Table 5-39
Scenario 5 CDR summary

DIALEDNO	CALLEDNO	OUTPULNO	TERMLRN
incoming IAM’s LRN	incoming IAM’s GAP	incoming IAM’s GAP	incoming IAM’s LRN
LNPCHECK	**ORIGLRN**	**PORTEDNO**	
6	JIP (if present) in IAM	empty	

Scenario 6: Processing of incoming LNP information followed by a NetworkBuilder query

When an LNP check has been performed on a previous switch (the incoming SS7 IAM message's FCI parameter has bit M set), the switch can launch a NetworkBuilder query and process the query's corresponding response.

Note 1: This scenario shows an incoming LRN followed by a NetworkBuilder query. However, it also provides information on functionality when the switch receives LNP information without an LRN and then launches a NetworkBuilder query.

Note 2: The CDR summary in this scenario provides information on the CDR population of additional scenarios involving incoming LNP information followed by a NetworkBuilder query.

Figure 5-32 illustrates Scenario 6. The following explains, in detail, Scenario 6:

1 The switch receives LNP information in the incoming IAM message. The IAM message contains the Forward Call Indicator (FCI) parameter with bit M set. If the IAM message contains an LRN, the

   — Generic Address Parameter (GAP) contains the directory number

   — Called Party Number (CPN) contains the LRN

2 The switch launches a NetworkBuilder query using an **`Info_Analyzed`** message. The **`Info_Analyzed`** message contains the ***`CalledPartyID`*** parameter and the `lnpReceived` extension parameter. The ***`CalledPartyID`*** parameter contains the directory number. The `lnpReceived` extension parameter notifies the SCP that the switch has received LNP information from a previous switch.

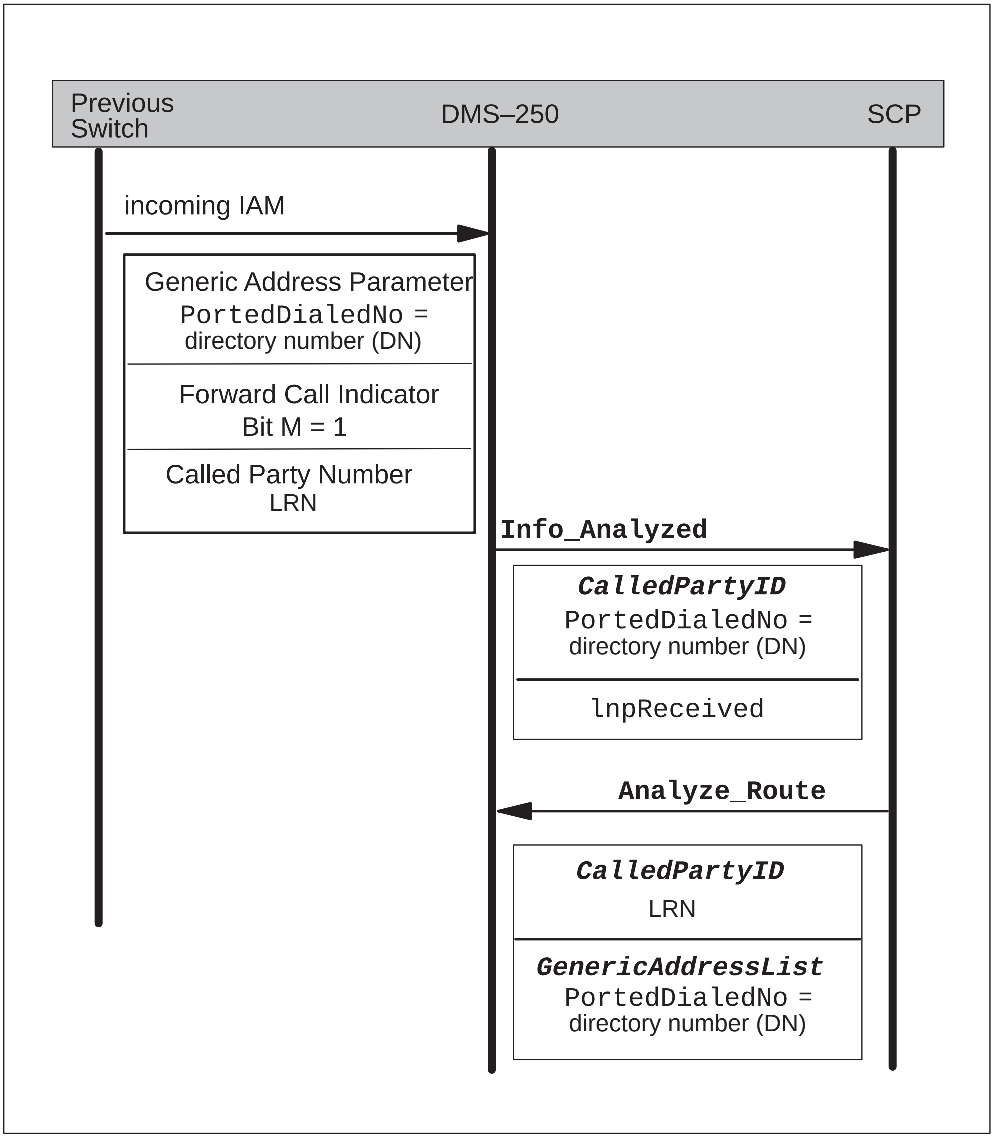
3 The SCP returns an **`Analyze_Route`** response message containing LNP information. The LNP information in the **`Analyze_Route`** response message overrides the LNP information the switch received in the incoming IAM message.

   If the SCP returns an LRN, the ***`CalledPartyID`*** parameter contains the LRN and the ***`GenericAddressList`*** parameter contains the `PortedDialedNo` (the directory number). (As shown in Figure 5-32.)

   Note: If the SCP does not return an LRN, the ***`CalledPartyID`*** parameter contains the directory number. (Not shown.)

Incoming LNP information (continued)

Figure 5-32
Scenario 6: Processing of incoming LNP information followed by a NetworkBuilder query



CDR summary

Table 5-40 summarizes the CDR population for six scenarios involving incoming LNP information followed by a NetworkBuilder query. Each scenario shows

- the parameters and information contained in the incoming IAM message
- the information contained in the NetworkBuilder SCP response message

Incoming LNP information (continued)

Table 5-40
Incoming IAM followed by a NetworkBuilder query CDR summary

Call Scenario	DIALEDNO field	CALLEDNO field	OUTPULNO field	PORTEDNO field
Incoming IAM: LRN NetworkBuilder SCP response: LRN	IAM LRN	SCP LRN	SCP LRN	SCP LNP GAP
Incoming IAM: Called Party Number, FCI NetworkBuilder SCP response: LRN	IAM Called Party Number	SCP LRN	SCP LRN	SCP LNP GAP
Incoming IAM: LRN NetworkBuilder SCP response: no LRN, but ***CalledPartyID***	IAM LRN	SCP ***CalledPartyID***	SCP ***CalledPartyID***	Empty
Incoming IAM: LRN NetworkBuilder SCP response: no LRN and no ***CalledPartyID***	IAM LRN	IAM LNP GAP	IAM LNP GAP	Empty
Incoming IAM: Called Party Number, FCI NetworkBuilder SCP response: no LRN, but ***CalledPartyID***	IAM Called Party Number	SCP ***CalledPartyID***	SCP ***CalledPartyID***	Empty
Incoming IAM: Called Party Number, FCI NetworkBuilder SCP response: no LRN and no ***CalledPartyID***	IAM Called Party Number	IAM Called Party Number	IAM Called Party Number	Empty

Incoming LNP information (continued)

Scenario 7: Processing of an incoming home LRN followed by a NetworkBuilder query

After the switch determines an incoming LRN is a home LRN, the switch can launch a NetworkBuilder query.

Figure 5-33 illustrates Scenario 7. The following explains, in detail, Scenario 7:

1 The switch receives an LRN in the incoming IAM message. The IAM message contains
   - — the Forward Call Indicator (FCI) parameter with bit M set
   - — the Called Party Number (CPN) parameter containing the LRN
   - — a Generic Address Parameter (GAP) containing the `PortedDialedNo`

2 The switch determines the LRN is a home LRN.

   Note: To determine whether the LRN is a home LRN, the switch verifies the LRN is entered in table TERMLRN.

3 The switch launches a NetworkBuilder query and the SCP returns a response message.

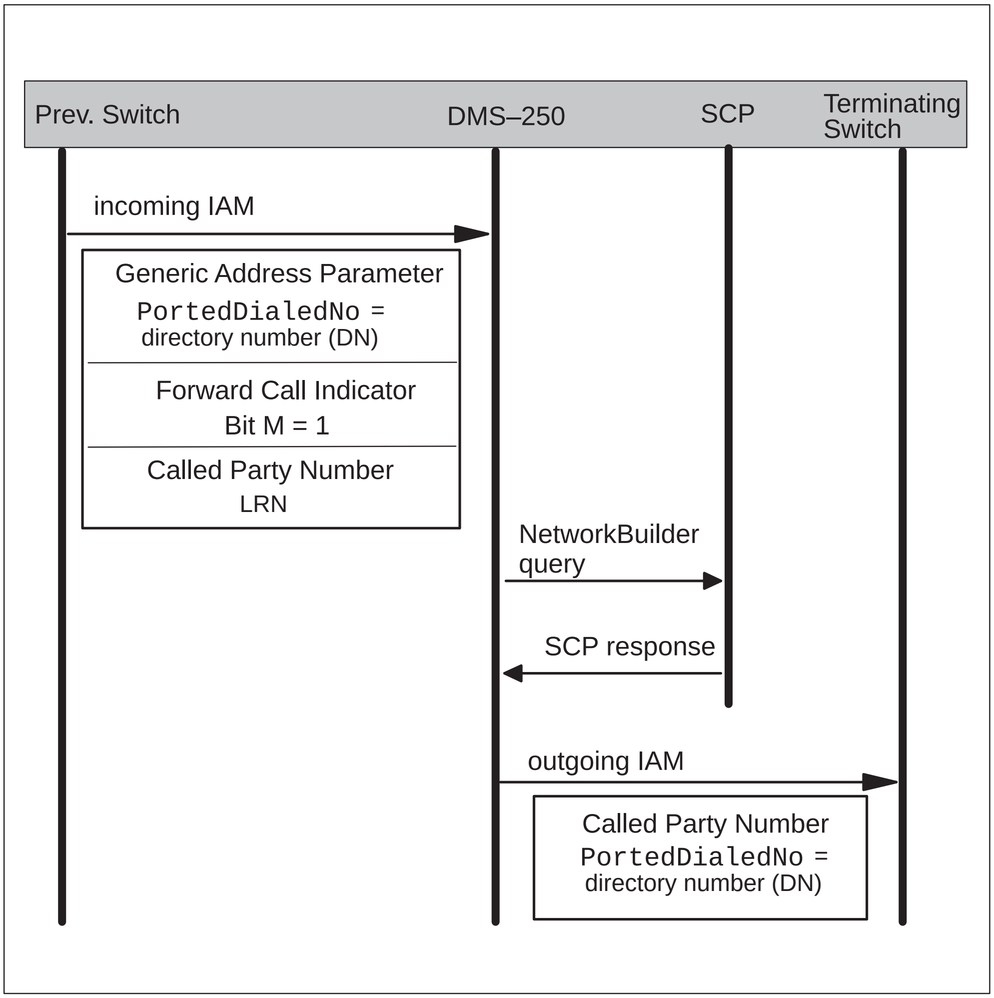
   Note 1: The `PortedDialedNo` in the Generic Address Parameter (GAP) is not present after the home LRN is processed, so it is not used in the **`Continue`**, **`Disconnect`**, **`Send_To_Resource`**, and error messages.

   Note 2: If the SCP returns an **`Analyze_Route`** message containing LNP information, the **`Analyze_Route`** message's LNP information overrides the LNP information received in the incoming IAM message.

4 The switch uses the `PortedDialedNo` to route the call and places the `PortedDialedNo` in the Called Party Number (CPN) in the outgoing IAM message.

Incoming LNP information (continued)

Figure 5-33
Scenario 7: Incoming Home LRN with NetworkBuilder query



Incoming LNP information (end)

OM information

The registers INLNPINF and TERMLRN are pegged.

CDR summary

Table 5-41 summarizes the CDR population for Scenario 7.

Table 5-41
Scenario 7 CDR summary

LNPCHECK	DIALEDNO	CALLEDNO	OUTPULNO
6	incoming IAM's GAP	SCP LRN	SCP LRN
CN1REQ	**TERMLRN**	**PORTEDNO**	
4	empty	empty	

Note: This table summarizes the CDR population for a scenario in which the SCP has returned an LRN.

Outgoing message parameters

ATTENTION

Outgoing messages require the CAIN0100 SOC option. Refer to *UCS DMS-250 NetworkBuilder Application Guide* for more information.

NetworkBuilder query messages used by NetworkBuilder are Advanced Intelligent Network (AIN) 0.2 TCAP messages. For Local Number Portability (LNP) services, however, NetworkBuilder may need to communicate with an LNP service control point (SCP) that is AIN 0.1 based. Therefore, the LNP **`Info_Analyzed`** message sent to the LNP SCP must conform to AIN 0.1 specifications.

The most distinguishing changes, from an LNP perspective, for AIN 0.1 messages to AIN 0.2 messages are the following:

- the ability to send extension parameters to the LNP SCP
- support for variations to the ***`UserID`*** parameter
- new ***`TriggerCriteriaType`*** parameters
- ***`BearerCapability`*** restrictions
- restrictions to the nature of address (NOA) and numbering plans for AIN digit type parameters

AIN 0.1 does not support extension parameters in messages sent to the SCP. Other important differences are listed in the parameter definitions in this chapter.

TDP-Request (query) messages

Once trigger criteria is met and extension blocks are allocated, the appropriate TCAP query message is built and sent to the LNP SCP. Table 6-1 lists all of the supported NetworkBuilder TDP-Request messages in the originating call model. The **`Info_Analyzed`** TDP-Request message is the only message specific to LNP.

Table 6-1
TDP-Request (query) messages

Query	Description
`Origination_Attempt`	Requests data from the SCP to direct call processing. This message is sent when a NetworkBuilder call meets the trigger criteria for an *Off_Hook_Immediate* trigger.
`O_Feature_Requested`	Requests data from the SCP to direct call processing. This message is sent when a NetworkBuilder call meets the trigger criteria for a *O_Feature_Requested* trigger.
`Info_Collected`	Requests data from the SCP to direct call processing. This message is sent when a NetworkBuilder call meets the trigger criteria for a *PRI_B-Channel* or *Shared_Interoffice_Trunk* trigger.
`Info_Analyzed`	Requests data from the SCP to direct call processing. This message is sent when a NetworkBuilder call meets the trigger criteria for a *Customized_Dialing_Plan*, *Specific_Digit_String,* or *Office_Code* trigger.
`Network_Busy`	Requests data from the SCP to direct call processing. This message is sent as a trigger detection point (TDP) request message when a NetworkBuilder call meets the trigger criteria for a *Network_Busy* trigger or as an EDP request message when a NetworkBuilder call encounters an active ***Network_Busy*** event detection point (EDP).
`O_Called_Party_Busy`	Requests data from the SCP to direct call processing. This message is sent as a TDP request message when a NetworkBuilder call meets the trigger criteria for a *O_Called_Party_Busy* trigger or as an EDP message when a NetworkBuilder call encounters an active ***O_Called_Party_Busy*** EDP.
Note: For more information on these messages, refer to the UCS DMS-250 NetworkBuilder Application Guide.	
—continued—	

Table 6-1
TDP-Request (query) messages (continued)

Query	Description
`O_No_Answer`	Requests data from the SCP to direct call processing. This message is sent as a TDP request message when a NetworkBuilder call meets the trigger criteria for a *O_No_Answer* trigger or as an EDP request message when a NetworkBuilder call encounters an active ***O_No_Answer*** EDP.
`O_Mid_Call`	Requests data from the SCP to direct call processing. This message is sent as a TDP request message when a NetworkBuilder call meets the trigger criteria for a *O_IEC_Reorigination* trigger.
Note: For more information on these messages, refer to the UCS DMS-250 NetworkBuilder Application Guide.	
—end—	

Messaging parameters

Table 6-2 lists the parameters that may be sent to the LNP SCP in an LNP **`Info_Analyzed`** TDP-Request message.

Table 6-2
Info_Analyzed TDP-Request message parameters

Parameter	Parameter
BearerCapability	***ChargePartyStationType***
CalledPartyID	***TriggerCriteriaType***
CallingPartyID	***UserID***
Carrier	***JurisdictionInfo***
ChargeNumber	

In addition, the LNP_PARAMETER_SET parameter in table CAINPARM allows the user to determine whether a subset of the LNP **`Info_Analyzed`** parameters will be included in the LNP **`Info_Analyzed`** message. COMPLETE_SET sends the following parameters:

- ***UserID***
- ***BearerCapability***
- ***CalledPartyID***
- ***TriggerCriteriaType***
- ***CallingPartyID***
- ***Carrier***
- ***ChargeNumber***
- ***ChargePartyStationType***
- ***JurisdictionInfo***

MINIMAL_SET sends the following parameters:

- ***UserID***
- ***BearerCapability***
- ***CalledPartyID***
- ***TriggerCriteriaType***

When a TDP-Request message is built, the switch formats the message and sends it to the message encoder. Once encoded, the message is sent to the LNP SCP.

Fatal application errors

Fatal application errors occur when NetworkBuilder call processing is unable to continue due to an unexpected error. Table 6-3 shows errors that may occur before an SCP response is returned:

Table 6-3
Fatal application errors

Error type	Log generated	Reported to SCP?	Error action performed	CDR COMPCODE
Message driver error (Note 1)	CAIN200	No	Default route (Note 2)	Note 3
CAIN_T1_TIMEOUT (Note 4)	CAIN200	Yes	Default route (Note 2)	9 (Note 5)
Unable to allocate extension block	CAIN200	No	Default route (Note 2)	Note 3
Unable to allocate VAMP resources	CAIN200 VAMP202	No	Default route (Note 2)	Note 3
Unable to allocate message against SOC option CAIN0100	CAIN102 CAIN200	No	Default route (Note 2)	Note 3

Note 1: A problem was encountered while sending the query message to the SCP.
Note 2: The error action for LNP is to default route. This is not datafillable.
Note 3: NetworkBuilder call processing does not set a call detail record (CDR) compcode. However, the compcode may be set by normal call processing
Note 4: The SCP did not respond within the provisioned amount of time.
Note 5: The CDR compcode may be overwritten by normal call processing.

Nonfatal application errors

Table 6-4 shows nonfatal application errors that can occur before an SCP response is returned:

Table 6-4
Nonfatal application errors

Error type	Log generated	Reported to SCP?	Error action performed
Failure to encode an optional parameter	CAIN101	No	Optional parameters are ignored

Caller abandon

Caller abandon occurs when the calling party disconnects while the switch is waiting for an SCP response. For more information concerning this error type, refer to the *UCS DMS-250 NetworkBuilder Application Guide*.

BearerCapability

Parameter definition

This parameter contains the bearer capability of the call at the time the query is sent.

Supported ***BearerCapability*** values are speech, 3_1 kHz audio, 56 kbps, and 64 kbps.

Note: 7 kHz audio and multirate, which are currently supported by CAIN 0.2, are not supported for an LNP query because they are not AIN 0.1 compliant.

Message types

The following TDP-Request (query) message supports the ***BearerCapability*** parameter:

- **Info_Analyzed**

Usage

Required

Range of values

speech
f3_1 kHz Audio
b56 kbps
b64 kbps

Population rules

None

Note: Refer to the *UCS DMS-250 NetworkBuilder AIN 0.2 TCAP Protocol Definition* for encoding information.

Restrictions

Supported ***BearerCapability*** values are speech, 3_1 kHz audio, 56 kbps, and 64 kbps.

If a value other than one of the valid values listed above is used, the query will be blocked. This is considered a fatal application error. A CAIN200 log is produced with the following error message: `Invalid bearer capability for an LNP query.`

BearerCapability (end)

Note: 7 kHz audio and multirate, which are currently supported by CAIN 0.2, are not supported for an LNP query because they are not AIN 0.1 compliant.

CalledPartyID

Parameter definition

The ***CalledPartyID*** is the known address or directory number of the called party. This number is used to route the call. When a call is initiated, ***CalledPartyID*** holds the dialed number. However, this number may change during standard pretranslation or N00 translation (in-switch or off-board).

AIN digits parameters are limited to use the AIN 0.1 Nature of Address (NOA) values and numbering plans. Specifically, valid NOA values are #3 – national (sig) number and #4 – international number. No change to the numbering plan values currently being used are required.

Message types

The following TDP-Request (query) message supports the ***CalledPartyID*** parameter:

- **Info_Analyzed**

Usage

Optional

Range of values

Maximum of 18 digits

Population precedence rules

Table 6-5 shows the population precedence rules for the ***CalledPartyID*** parameter.

Table 6-5
CalledPartyID parameter population rules

	Rule	NOA	Numbering Plan
1	International or international partitioned numbers	INTL	ISDN
2	OFFNET	NATL	ISDN

Note: Refer to the *UCS DMS-250 NetworkBuilder AIN 0.2 TCAP Protocol Definition* for encoding information.

CalledPartyID (end)

Restrictions

The content of this parameter is used for global title with CAIN_OFCD_GT global title translations.

Parameter definition

The ***CallingPartyID*** contains the address used for caller identification purposes and is always transmitted.

AIN digits parameters are limited to use the AIN 0.1 NOA values and numbering plans. Specifically, valid NOA values are #3 – national (sig) number and #4 – international number. No change to the numbering plan values currently being used are required.

Message types

The following TDP-Request (query) message supports the ***CallingPartyID*** parameter:

- **Info_Analyzed**

Usage

Optional

Range of values

Maximum of 24 digits

Population precedence rules

For information on the population precedence rules for the ***CallingPartyID*** parameter, refer to the *UCS DMS-250 NetworkBuilder Application Guide*.

Restrictions

- PANIs are not supported.
- This parameter is always sent with presentation allowed.
- The presentation restriction indicator from an incoming SS7 IAM or PRI SETUP message is ignored.

Carrier (end)

Parameter definition

This parameter is used to transmit the carrier identification code (CIC) received from an access tandem switch, when available, or when parameter SEND_CARRIER_FROM_TRKGRP is set to Y.

No changes were made to this parameter for LNP. For more information regarding this parameter, refer to the *UCS DMS-250 NetworkBuilder Application Guide*.

Parameter definition

This parameter holds the billing number that would be used to populate the CDR at this point in the call. If the switch has not determined a charge number, this parameter is not transmitted to the SCP.

Message types

The following TDP-Request (query) messages support the ***ChargeNumber*** parameter:

- **Info_Analyzed**

Usage

Optional

Note: This parameter is not used in EDP request messages.

Range of values

Maximum of 24 digits

Population precedence rules

Table 6-6 shows the population precedence rules for the ***ChargeNumber*** parameter as they relate to LNP.

Table 6-6
ChargeNumber parameter population rules

	Rule	NOA	Numbering Plan
1	ANI billed	INTL INTL UNK	PVT
Note: Refer to the *UCS DMS-250 NetworkBuilder AIN 0.2 TCAP Protocol Definition* for encoding information.			

Restrictions

None

ChargePartyStationType (end)

Parameter definition

This parameter contains the information digits from the call.

Population precedence rules

The CAIN_PROTOCOL_VERSION parameter does not affect the population of the ***ChargePartyStationType*** parameter for OFFCCODE.

Note: Refer to *UCS DMS-250 NetworkBuilder Application Guide* for more information on the ***ChargePartyStationType*** parameter.

TriggerCriteriaType

Parameter definition

This parameter holds the nature of the criteria used to trigger and allows the trigger to be identified.

Bellcore has added a new trigger criteria value of 27, LNP_OFCD, for use with the *Office_Code* trigger. Meanwhile, ICC has specified that NPA, NPA_NXX, NPA_NXXX, NPA_NXXXX, NPA_NXXXXX, or NPA_NXXXXXX could be used. CAIN 0.2 currently supports NPA, NPA_NXX, and NPA_NXXXXXX. The other ***`TriggerCriteriaType`*** values specified by ICC are reserved and are not required for AIN 0.1. However, they are supported by LNP. The trigger criteria values are datafillable in table OFFCCODE.

Note: Currently, NetworkBuilder denotes NPA_NXXXXXX, value #8, as SDS_ADDR. There are no software problems foreseen as a result. However, tools designed to display this value, for example, a VAMP902 log, still displays SDS_ADDR, regardless of whether it was used for LNP.

Message types

The following TDP-Request (query) message supports the ***`TriggerCriteriaType`*** parameter:

- **`Info_Analyzed`**

Usage

Required

Range of values

Refer to population rules.

Population rules

Table 6-7 shows the population rules for the ***`TriggerCriteriaType`*** parameter.

TriggerCriteriaType (end)

Table 6-7
TriggerCriteriaType parameter population rules

Rule	TriggerCriteriaType value
Office_Code trigger with NPA address	NPA
Office_Code trigger with NPA_NXX address	NPA_NXX
Office_Code trigger with NPA_NXXX address	NPA_NXXX
Office_Code trigger with NPA_NXXXX address	NPA_NXXXX
Office_Code trigger with NPA_NXXXXX address	NPA_NXXXXX
Office_Code trigger with NPA_NXXXXXX address	NPA_NXXXXXX
Office_Code trigger with LNP_OFCD address	LNP_OFCD
Note: Refer to the *UCS DMS-250 NetworkBuilder AIN 0.2 TCAP Protocol Definition* for encoding information.	

Restrictions

Currently, NetworkBuilder denotes NPA_NXXXXXX, value #8, as SDS_ADDR. There are no software problems foreseen as a result. However, tools designed to display this value, for example, a VAMP902 log still displays SDS_ADDR, regardless of whether it was used for LNP.

Note: If the VAMP902 log is used to view this TDP-Request message, the text string for value NPA_NXXXXXX will appear as SDS_ADDR.

UserID

Parameter definition

The CAIN 0.2 ***UserID*** parameter supports a switch ID (SWID) and trunk group number prefixed by two zeros which are combined and encoded in the directory number (DN) format as specified in the *UCS DMS-250 NetworkBuilder AIN 0.2 TCAP Protocol Definition*. LNP uses two of the formats designated for the ***UserID*** parameter. The two formats that are used are either a true DN (10-digit ANI/CLID) or a trunk group ID with the trunk group number from table CLLI (4 digits). Refer to *Advanced Intelligent Network (AIN) 0.1 Switch—Service Control Point (SCP) Application Protocol Interface Generic Requirements*, TR-NWT-001285 for more information.

For LNP queries, the ***UserID*** parameter takes the DN or TRK form.

Message types

The following TDP-Request (query) message supports the ***UserID***

- **Info_Analyzed**

Usage

Required

Range of values

The following table shows the range of values available for the AIN 0.1 ***UserID*** parameter.

Table 6-8
AIN 0.1 UserID parameter values

Parameter	Range of values
DN	10–digit ANI/CLID
TRK	0–8191

Note: For information regarding the range of values available for the CAIN 0.2 ***UserID*** parameter, refer to the *UCS DMS-250 NetworkBuilder Application Guide*.

UserID (end)

Population rules

Currently, call processing currently only populates this parameter using the TRK form.

Note: Refer to the *UCS DMS-250 NetworkBuilder AIN 0.2 TCAP Protocol Definition* for encoding information.

Restrictions

None

Parameter definition

This parameter holds the originating switch's location routing number (LRN).

Note: The ***`JurisdictionInfo`*** parameter is controlled through the LNP_PROTOCOL_STREAM office parameter in table CAINPARM.

Message types

The following TDP-Request (query) message supports the ***`JurisdictionInfo`*** parameter:

- **`Info_Analyzed`**

Usage

Optional

Range of values

Maximum of 10 digits

Population rules

If an incoming ***`JurisdictionInfo`*** parameter was received on an incoming IAM, it is used. Otherwise, the default JIP value from table TRKGRP, if present, is used.

Restrictions

None

Incoming message parameters

SCP responses

When the switch receives a message from the service control point (SCP), NetworkBuilder processes the response. Response processing consists of interpreting, validating, and performing the instructions in the SCP Transaction Capabilities Application Part (TCAP) message. Table 7-1 lists the SCP response messages

Table 7-1
SCP Response messages

Response	Description
`Analyze_Route`	This message instructs the switch to resume NetworkBuilder call processing, using the address, billing, and routing information provided by the SCP.
`Continue`	This message instructs the switch to route the call using in-switch routing information and perform as if the call had not triggered a query to the SCP.
`Disconnect`	This message instructs the switch to disconnect the call and apply treatment.
`Send_To_Resource` or **`Connect_To_Resource`** in a Response package	This message is ignored, and default routing occurs for LNP.
`Send_To_Resource` in a Conversation package	A **`Resource_Clear`** message in a Response package is sent.
Note 1: The switch can receive additional SCP response messages that don't interact with LNP. For information on the above messages as they relate to NetworkBuilder or additional SCP response messages, refer to the *UCS DMS-250 NetworkBuilder Application Guide*. and the *UCS DMS-250 NetworkBuilder AIN 0.2 TCAP Protocol Definition*. ***Note 2:*** The **`Connect_To_Resource`** with ***`DestinationAddress`*** parameter is used only at the ***O_Mid_Call*** TDP and EDP.	
—continued—	

Table 7-1
SCP Response messages (continued)

Response	Description
Connect_To_Resource in a Conversation package	A **CTR_Clear** message in a Response package is sent
Send_To_Resource with ***DestinationAddress*** parameter	Instructs the switch to route the call to an intelligent peripheral (IP)
Connect_To_Resource with ***DestinationAddress*** parameter	Instructs the switch to route the call to an intelligent peripheral (IP) (Note 2)
Request_Report_BCM_Event non-call-related component	Instructs the switch to arm EDPs

Note 1: The switch can receive additional SCP response messages that don't interact with LNP. For information on the above messages as they relate to NetworkBuilder or additional SCP response messages, refer to the *UCS DMS-250 NetworkBuilder Application Guide*. and the *UCS DMS-250 NetworkBuilder AIN 0.2 TCAP Protocol Definition*.
Note 2: The **Connect_To_Resource** with ***DestinationAddress*** parameter is used only at the ***O_Mid_Call*** TDP and EDP.

—end—

Error messages

The switch can receive the following error messages from the SCP:

- **Application_Error**
- **Failure_Report**
- **Report_Error**

Analyze_Route message
LNP query (continued)

The information contained in the `Analyze_Route` message may differ slightly depending on whether the message is in response to a Local Number Portability (LNP) query or a NetworkBuilder query. Therefore, this section contains information on both scenarios.

Use

In general, the `Analyze_Route` message instructs the switch to resume NetworkBuilder call processing, using the address, billing, and routing information provided by the SCP. There are two types of routing: direct termination and standard. Only standard routing is supported for LNP queries. For information on direct termination routing, refer to the *UCS DMS-250 NetworkBuilder Application Guide*.

Note: The SCP may include the `Request_Report_BCM_Event` non-call related component with the `Analyze_Route` message in a conversational TCAP package. The message is not expected to contain this component. However, if this component is present, the transaction is closed and a `Close` message with a ***`CloseCause`*** value of `unexpectedCommunication` is sent.

Terminology

The following terms are used to define NetworkBuilder routing:

- **Route indexes** are determined by the data provided in an `Analyze_Route` response or through normal translations. The index points to the appropriate table that identifies a routing list.
- **Route lists** contain the routes to be attempted by the switch when establishing the call. Route lists are provisioned in tables, such as: TANDMRTE, TERMRTE, HNPACONT, FNPACONT, CTRTE, and OFRT. Each route list can contain multiple routes.
- **Route** typically consists of a route selector, connection type, and clli (common language location identifier).
- **CAIN routing parameters** are provided in an `Analyze_Route` response. The only parameter that applies to LNP is: ***`CalledPartyID`***.

  Note: In conjunction with standard routing parameters, ***`CalledPartyID`*** requires in-switch translations to derive the route index. One route index is calculated for each parameter.

Analyze_Route message
LNP query (continued)

Standard routing

The switch normally routes a call using standard routing. Call processing analyzes the dialed address along with other call data (such as STS and call type) to determine a route index.

Call processing selects the translation table based on the nature of address, as follows:

- national NOA – table HNPACONT
- international NOA – table CCTR
- international partitioned NOA – table STS2CCDB, CTHEAD, CTCODE, CTRTE

Message parameters

Table 7-2 lists the parameters for the Analyze_Route message.

Table 7-2
Analyze_Route parameters

Parameter	Definition
CalledPartyID	This parameter contains the translated address of the called party or the location routing number (LRN).
GenericAddressList	
AlternateOutpulseNo	This extension parameter contains digits to be outpulsed when the ***AlternateTrunkGroup*** number to outpulse field is 0.
SecondAlternateOutpulseNo	This extension parameter contains the digits to be outpulsed when the ***SecondAlternateTru***nkGroup number to outpulse field is 0.
OverflowRoutingNo	This extension parameter contains the overflow routing number.
DialedNoInwardService	This extension parameter contains the DNIS value specific to the Dialed Number Inward Service specified.
—continued—	

Analyze_Route message
LNP query (continued)

Table 7-2
Analyze_Route parameters (continued)

Parameter	Definition
`PortedDialedNo`	Contains the original called party address of a successful LNP query
`ForwardCallIndicator`	Indicates that an LNP check has been performed at the SCP
—end—	

Parameter processing

The following parameters returned in an **`Analyze_Route`** message, received in response to a NetworkBuilder query, are saved prior to a subsequent LNP query being performed. This prevents critical call-related data from the original **`Analyze_Route`** message from being lost when the SCP returns an **`Analyze_Route`** message in response to the LNP query.

- ***`CalledPartyID`***
- ***`CallingPartyID`***
- ***`ChargeNumber`***
- ***`Carrier`***
- ***`AlternateCarrier`***
- ***`SecondAlternateCarrier`***
- ***`AMADigitsDialedWC`***
- ***`ExtensionParameter`***
  - `servTranslationScheme`
  - `reorigAllowed`
  - `satRestriction`
  - `classOfSvc`

  Note: No class of service screening should be performed on the LRN that is returned in response to an LNP query.

  - `cainGroup`
  - `billSequenceNumber`
  - `netinfo`

Analyze_Route message
LNP query (continued)

— `callBranding`

- ***GenericAddressList***

— `OverflowRoutingNo`

— `DialedNoInwardService`

NetworkBuilder call processing uses steps in Table 7-3 to process the parameters in an **Analyze_Route** message:

Table 7-3
Parameter processing steps

Step	Call Processing Action
1	Check for ***CalledPartyID***. If the ***CalledPartyID*** parameter is present, it is compared to the ***CalledPartyID*** parameter sent to the SCP in the query. If the values are different, the ***CalledPartyID*** is treated as the LRN. If the values are the same, the call is routed based on the Called Party Address.
2	If a CAIN 0.2 only parameter is received, a non–fatal application error occurs.
3	If a CAIN 0.2 parameter is received that is also in the AIN 0.1 protocol, then the parameter is decoded but is ignored by LNP. If the switch receives a parameter that is valid for both CAIN 0.2 and AIN 0.1, but the parameter is in the CAIN 0.2 format, then NetworkBuilder decodes the parameter and LNP ignores the parameter.
4	If the switch receives an AIN 0.1 parameter that NetworkBuilder does not support, then a non-fatal application error occurs.
5	If a ***CalledPartyID*** is not returned, default routing takes place.

Nonfatal application errors

Table 7-4 lists errors that can occur while the **`Analyze_Route`** parameters are being processed.

Table 7-4
Analyze_Route nonfatal application errors

Error type	Log generated	Reported to SCP?	Error action performed
A CAIN 0.2 parameter is received	CAIN100	No	Parameters are ignored
An AIN 0.1 only parameter that NetworkBuilder does not support is received	CAIN101	No	Parameters are ignored
No routing parameter available	None	No	Default route. This is not LNP specific; it is specific to the **`Info_Analyzed`** trigger detection point (TDP).

Associated logs

CAIN100, CAIN102

Note: For more information on CAIN100 logs, refer to the *UCS DMS-250 Logs Reference Manual*.

Associated OMs

CAINMSGR, CAINAGOM, CAINTRIG, CAINLNP

Note: For more information on NetworkBuilder OMs, refer to the *UCS DMS-250 Logs Reference Manual*.

Analyze_Route message NetworkBuilder query (continued)

Use

In general, the **`Analyze_Route`** message instructs the switch to resume NetworkBuilder call processing, using the address, billing, and routing information provided by the SCP. There are two types of routing: direct termination and standard termination. For information on NetworkBuilder routing, refer to the *UCS DMS-250 NetworkBuilder Application Guide*.

Note: The SCP may include the **`Request_Report_BCM_Event`** non-call related component with the **`Analyze_Route`** message in a conversational TCAP package.

Terminology

The following terms are used to define NetworkBuilder routing:

- **Route indexes** are determined by the data provided in an **`Analyze_Route`** response or through normal translations. The index points to the appropriate table that identifies a routing list.
- **Route lists** contain the routes to be attempted by the switch when establishing the call. Route lists are provisioned in tables, such as: TANDMRTE, TERMRTE, HNPACONT, FNPACONT, CTRTE, and OFRT. Each route list can contain multiple routes.
- **Routes** typically consists of a route selector, connection type, and common language location identifier (CLLI).
- **CAIN routing parameters** are provided in an **`Analyze_Route`** response. The parameters are: ***`PrimaryTrunkGroup`***, ***`AlternateTrunkGroup`***, ***`SecondAlternateTrunkGroup`***, ***`Carrier`***, ***`AlternateCarrier`***, ***`SecondAlternateCarrier`***,***`CalledPartyID`***, and the ***`GenericAddressList`*** parameter's `OverflowRoutingNo`.

  Note 1: Parameters ***`PrimaryTrunkGroup`***, ***`AlternateTrunkGroup`***, and ***`SecondAlternateTrunkGroup`*** are used to identify direct termination route indexes into tables TANDMRTE and TERMRTE.

  Note 2: Standard routing parameters, ***`Carrier`***, ***`AlternateCarrier`***, and ***`SecondAlternateCarrier`*** are used to identify a route by obtaining an STS value through table CICROUTE.

  Note 3: Standard routing parameters, ***`CalledPartyID`*** and/or the `servTranslationScheme` (or `univIdx` for SS7 Global-IMTs), and ***`GenericAddressList`*** parameter's `OverflowRoutingNo` require in-switch translations to derive the route index. One route index is calculated for each parameter.

Analyze_Route message NetworkBuilder query (continued)

- **Serving translation scheme (STS)** extension parameters are provided in an **`Analyze_Route`** response along with CAIN routing parameters. Each CAIN routing parameter has an associated STS extension parameter. Default STS extension parameters are also datafilled in table CAINXDFT to be used when the SCP fails to return the associated STS extension parameter with the CAIN routing parameter. Additionally, the original STS derived prior to the query message may be used when the STS extension parameter is not returned from the SCP and is not datafilled in table CAINXDFT. Figure 7-1 illustrates the order of precedence used to obtain the STS for each CAIN routing parameter:

Figure 7-1
Precedence order for STS extension parameters

STS Extension Parameters / Route Parameters Returned by the SCP	PRISTS	ALTSTS	SALTSTS	STS	OVLFSTS	table CAINXDFT PRISTS	table CAINXDFT ALTSTS	table CAINXDFT SALTSTS	table CAINXDFT STS	table CAINXDFT OVLFSTS	Pre Query STS
`PrimaryTrunkGroup`	1			3		2			4		5
`AlternateTrunkGroup`		1		3			2		4		5
`SecondAlternateTrunkGroup`			1	3				2	4		5
`CalledPartyID`				1					2		3
`OverflowRoutingNo`				3	1				4	2	5

LEGEND

`PRISTS (primaryTrunkGroupSTS)` – STS to be used with the ***`PrimaryTrunkGroup`*** parameter

`ALTSTS (alternateTrunkGroupSTS)` – STS to be used with the ***`AlternateTrunkGroup`*** parameter

`SALTSTS (secondAlternateTrunkGroupSTS)` – STS to be used with the ***`SecondAlternate TrunkGroup`*** parameter

`STS (servTranslationScheme)` – STS to be used with the ***`CalledPartyID`*** parameter

`OVFLSTS (overflowRoutingNo)` – STS to be used with the `OverflowRoutingNo` parameter

Note: The `univIdx` extension parameter is used to specify the STS for SS7 Global-IMTs.

Analyze_Route message NetworkBuilder query (continued)

- **Standard route advance –** The switch attempts the first route in route list, and when unable to terminate using the route, attempts to establish the call using the next route in the route list.
- **CAIN routing parameter advance –** The switch attempts to route according to data provided in the next untried CAIN routing parameter. When unable to establish the connection using the data, call processing advances to the next CAIN parameter to derive a new route list and attempts to route the call.

An **`Analyze_Route`** message contains the routing information used to terminate the call. The **`Analyze_Route`** message may contain up to eight parameters to derive routing indexes. The precedence is as follows:

1. Direct termination routing using the ***`PrimaryTrunkGroup`*** parameter.
2. Direct termination routing using the ***`AlternateTrunkGroup`*** parameter.
3. Direct termination routing using the ***`SecondAlternateTrunkGroup`*** parameter.
4. Standard routing using the ***`Carrier`*** parameter.
5. Standard routing using the ***`AlternateCarrier`*** parameter.
6. Standard routing using the ***`SecondAlternateCarrier`*** parameter.
7. Standard routing using the ***`CalledPartyID`*** parameter and/or the `servTranslationScheme` (or `univIdx` for SS7 Global-IMTs).
8. Standard routing using the ***`GenericAddressList`*** parameter's `OverflowRoutingNo` and `overflowRoutingNoSTS`.

Each of these routing choices leads to the determination of a route list which is used to attempt termination. If translations are not successful with these route choices, NetworkBuilder provides the ability to continue through each of the received indexes without further consultation of the SCP.

Note: Direct termination routing does not analyze dialed addresses.

Route advancing

Each of these routing choices leads to the determination of a route list which is used to attempt termination. If translations are not successful with individual routing parameters, NetworkBuilder provides the ability to continue through each of the received indexes without further consultation of the SCP or trigger tables.

Analyze_Route message
NetworkBuilder query (continued)

Route advancing can be performed and controlled either through the provisioning of *Network_Busy*, *O_Called_Party_Busy*, and *O_No_Answer* triggers, or through standard route advancing. Following are descriptions of the busy conditions which are considered by NetworkBuilder processing:

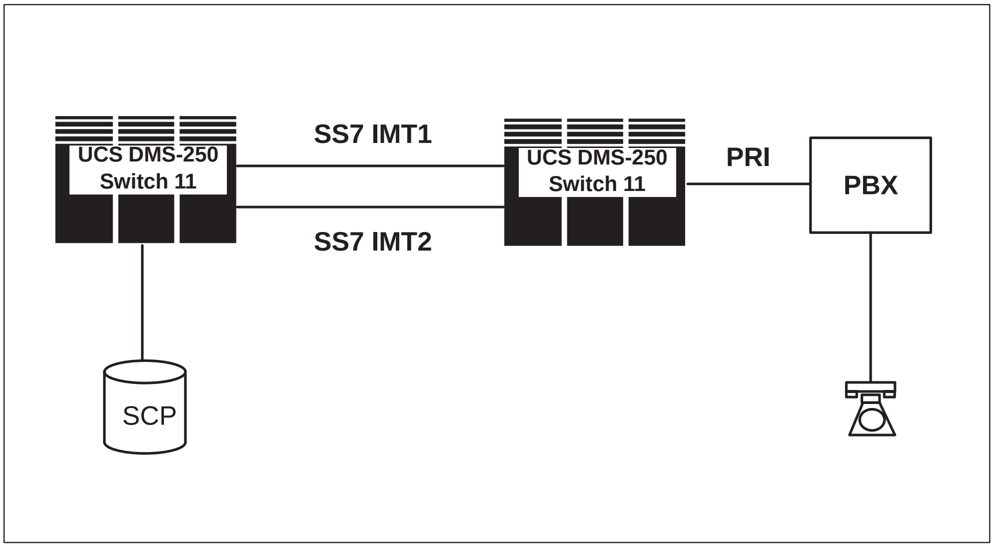
- Route Busy – Occurs when attempting to route a call over a trunk group which is busy at the SSP which queried.
- Network Busy – Occurs in each of the following scenarios:
  - — When the Route Busy condition occurs and no additional routes are available at the querying SSP.
  - — The querying SSP receives a release cause from the terminating trunk indicating that a route was busy at a switch other than the SSP. This requires PRI or SS7 connectivity.
- User Busy – Occurs in each of the following scenarios:
  - — The querying SSP attempts to route to a DAL or AXXESS trunk with ONNETTRK = Y which is busy. If ONNETTRK = N, the Route Busy condition is detected.
  - — The querying SSP receives a release cause from the terminating trunk indicating that the end user was busy. This requires PRI or SS7 connectivity.
- No Answer – Occurs when the No Answer timer has expired on the querying SSP.

Encountering the Network Busy, User Busy, and No Answer scenarios can be precursors to encountering the Route busy scenario when multiple route choices are available. Thus, either of the Network Busy, User Busy, or No Answer scenarios could be encountered and acted on according to the detection point evaluation resulting in a route advancing action. If this route attempt is unable to leave the querying SSP, the Route Busy condition would then be detected again, route advancing could occur.

Analyze_Route message
NetworkBuilder query (continued)

Figure 7-2 provides a network configuration diagram that can be used as an example for each of the Busy conditions.

Figure 7-2
Network Configuration



1 Route Busy – Occurs when the SSP is attempting to route to the phone, but the SS7 IMT1 trunk is busy.

2 Network Busy – Occurs when the SSP is attempting to route to the phone and both SS7 IMT trunks are busy.

3 User Busy – Occurs when the SSP is attempting to route to the phone, but the phone is busy.

4 No Answer – Occurs when there is successful termination to the phone, but there is no answer detected during the time allotted by the No Answer timer.

The action associated with *Network_Busy*, *O_Called_Party_Busy*, and *O_No_Answer* triggers may direct the call to route advance based on the evaluation of the current routing condition at the time the detection point is encountered.

- If there are unattempted route list choices or CAIN routing parameters, the routing condition criteria is RTEAVAIL.
- If there are no unattempted route list or CAIN routing parameters, the routing condition criteria is RTESDONE.

Analyze_Route message NetworkBuilder query (continued)

- If there are additional route list or CAIN routing parameters and a release cause of TERM_RESOURCE_UNAVAILABLE was received from the network, the routing condition criteria is TERMRTE_GNCT.

The type of route advancing performed depends on the action associated with the detection point, either NEXTRTE or NEXTCNRTE.

An action of NEXTRTE causes the current route list to be completed and if possible, to route advance through the CAIN routing parameters in the following order:

— ***PrimaryTrunkGroup***

— ***AlternateTrunkGroup***

— ***SecondAlternateTrunkGroup***

— ***Carrier***

— ***AlternateCarrier***

— ***SecondAlternateCarrier***

— ***CalledPartyID*** and/or the `servTranslationScheme` (or `univIdx` for SS7 Global-IMTs).

— ***GenericAddressList*** parameter's `OverflowRoutingNo`. and `overflowRoutingNoSTS`

An action of NEXTCNRTE causes the call to stop attempting to process the current route list and proceed to the next CAIN routing parameter. However, the carrier parameters and the called party are treated as a single routing choice. Therefore, the parameters are processed in this order.

— ***PrimaryTrunkGroup***

— ***AlternateTrunkGroup***

— ***SecondAlternateTrunkGroup***

— ***Carrier***, ***AlternateCarrier***, ***SecondAlternateCarrier*** and ***CalledPartyID*** and/or the `servTranslationScheme` (or `univIdx` for SS7 Global-IMTs).

— ***GenericAddressList*** parameter's `OverflowRoutingNo`.

Since the routing condition criteria is part of the key for determining the appropriate action for the detection points, only one action may be associated with each the RTEAVAIL and RTESDONE criteria. There is an implied dependency on this provisioning limitation which links the

Analyze_Route message NetworkBuilder query (continued)

provisioned action for the RTEAVAIL tuple to the determination of the RTESDONE criteria.

When the RTEAVAIL tuple action is NEXTRTE, the RTESDONE condition can only be achieved by completely exhausting all route possibilities (route list and NetworkBuilder). When the RTEAVAIL tuple action is NEXTCNRTE, the RTESDONE criteria is achieved once each of the provided CAIN routing parameters are attempted (even though additional route list choices may still be available).

Direct termination routing

NetworkBuilder allows the switch to serve as an originating, intermediary, or terminating switch. The originating switch has access to the following:

- originating switch-based virtual private network (VPN) data
- connected switching facilities data (through CCS7 TCAP transactions)

When a call originates from an agent carrying a NetworkBuilder call, the originating switch passes the called number to the SCP through a CCS7 TCAP transaction. The SCP returns a destination switch identifier (SWID) and a destination trunk group number.

Note: Every switch in the network must be assigned a unique SWID.

The switch performs direct termination routing through the use of two tables: TANDMRTE (Tandem Routing) and TERMRTE (Termination Routing). Table TANDMRTE directs the call from switch to switch until the terminating switch in the network is reached. Then, table TERMRTE directs the call to the terminating route index.

Note 1: The originating switch may also be the terminating switch, in which case table TANDMRTE is not used.

Note 2: SS7 Global-IMTs should use the ADDR dialing plan for direct termination routing through table TANDMRTE.

Limitations and restrictions

Direct termination through a tandem switch (using table TANDMRTE) is only supported over IMTs that are datafilled to support UCS-to-UCS ISUP protocol. When any other protocol is used, retranslation occurs at the tandem switch.

Analyze_Route message NetworkBuilder query (continued)

Table routing

The `Analyze_Route` message may contain the following routing information in ***PrimaryTrunkGroup***, ***AlternateTrunkGroup***, or ***SecondAlternateTrunkGroup*** parameters used for direct termination routing:

- destination switch ID (SWID)
- destination trunk group number

Call processing checks the destination SWID against the current switch's SWID (located in table OFCVAR, parameter ORIG_SWITCH_ID). If the values are equal, the originating switch is also the terminating switch and table TERMRTE directs the routing.

However, if the SWID values are not equal, the destination SWID indexes table TANDMRTE to access the intermediary routing switch. The destination SWID and trunk group information is encoded into an ISUP generic digits parameter (GDP) of the initial address message (IAM) delivered to the intermediary switch.

The Called Party Number of the IAM may contain one of the following parameters returned from the SCP (in order of precedence):

1 appropriate outpulse number for the selected route choice is from the

- — ***PrimaryTrunkGroup***, contains the ***OutpulseNumber***
- — ***AlternateTrunkGroup***, contains the `AlternateOutpulseNo` (from the ***GenericAddressList***, when available) or the ***OutpulseNumber***
- — ***SecondAlternateTrunkGroup***, contains the `SecondAlternateOutpulseNo` (from the ***GenericAddressList***, when available) or the ***OutpulseNumber***

2 ***CalledPartyID*** (returned from the SCP)

Note: If an SS7 Global-IMT has not used the ADDR dialing plan, additional digits are included., based on the dialing plan used. Refer to the *UCS DMS-250 International Application Guide* for more information on outpulsing digits for SS7 Global-IMTs.

3 original value of ***CalledPartyID*** (delivered to the SCP in the query)

When the tandem switch receives the IAM message, the SWID in the GDP is checked against this switch's ORIG_SWITCH_ID parameter. If the destination SWID does not equal tandem switch's identifier, table

Analyze_Route message NetworkBuilder query (end)

TANDMRTE is again indexed and the call is routed to another tandem switch.

Tandem routing continues until the terminating SWID equals the tandem switch's ORIG_SWITCH_ID. Table TERMRTE is then indexed and the call is routed to the appropriate agency. The digits outpulsed to the terminating agency are the Called Party Number from the IAM.

Analyze_Route message
NetworkBuilder query (continued)

Standard routing

The switch normally routes a call using standard routing. Call processing analyzes the dialed address along with other call data (such as STS and call type) to determine a route index.

When the switch receives an **`Analyze_Route`** from the SCP, call processing attempts to route the call using the following parameters in the following order:

1. ***`PrimaryTrunkGroup`***
2. ***`AlternateTrunkGroup`***
3. ***`SecondAlternateTrunkGroup`***
4. ***`Carrier`***
5. ***`AlternateCarrier`***
6. ***`SecondAlternateCarrier`***
7. ***`CalledPartyID`***
8. ***`GenericAddressList`***'s `OverflowRoutingNo`

The switch attempts to route the first three parameters. If those parameters are not returned in the message, the next routing index (through normal switch translations) is determined by using the ***`Carrier`***, ***`AlternateCarrier`***, ***`SecondAlternateCarrier`***, ***`CalledPartyID`*** or the ***`GenericAddressList`***'s `OverflowRoutingNo` and/or the `servTranslationScheme`.

Call processing selects the translation table based on the nature of address, as follows:

- national NOA – table HNPACONT
- international NOA – table CCTR
- international partitioned NOA – table STS2CCDB, CTHEAD, CTCODE, CTRTE

Note: If an STS extension parameter was returned with it's associated routing parameter in the **`Analyze_Route`**, call processing uses this new STS for translations. If the SCP does not return an STS, call processing uses the default STS from table CAINXDFT (if available). If a default STS is not provisioned, the switch uses the STS related to the call before the query was sent. Figure 2-1, shown previously, illustrates the order of precedence used to obtain the STS for each CAIN routing parameter.

Analyze_Route message NetworkBuilder query (continued)

Message parameters

Table 7-5 lists the parameters that may be returned by the SCP:

Table 7-5
Analyze_Route parameters

Parameter	Definition
ChargeNumber	Contains the billing number
CallingPartyID	Contains the address used for caller identification purposes
ChargePartyStationType	Contains the information digits for the call
CalledPartyID	Contains the translated address of the called party or the LRN
OutpulseNumber	Contains the digits to be outpulsed when the ***PrimaryTrunkGroup*** Number to Outpulse field is 0. This number is used when the outpulse number associated with ***AlternateTrunkGroup*** or ***SecondAlternateTrunkGroup*** is not available.
PrimaryTrunkGroup	Contains the SCP-determined primary route index for table TERMRTE or TANDMRTE
AlternateTrunkGroup	Contains the SCP-determined alternate route index for table TERMRTE or TANDMRTE
SecondAlternateTrunkGroup	Contains the SCP-determined second alternate route index for table TERMRTE or TANDMRTE
Carrier	Contains the carrier selection information and the carrier identification code (CIC) to route the call
AlternateCarrier	Contains the alternate carrier selection information and the CIC to route the call
SecondAlternateCarrier	Contains the second alternate carrier selection information and the CIC to route the call
AMAAlternateBillingNumber	Identifies an alternate billing number to which the AIN service should be billed. The switch uses the contents of this parameter to populate the ALTBILL CDR field.
AMALineNumber	Contains digits used to populate one or more of the following CDR fields: ANISP, CLGPTYNO, or PRESIND

—continued—

Analyze_Route message
NetworkBuilder query (continued)

Table 7-5
Analyze_Route parameters (continued)

Parameter	Definition
AMAslpID	Contains a digit string to be populated into the SLPID CDR field
AMADigitsDialedWC	Contains digit strings to be populated into one or more of the following call detail record fields: PINDIGS, ACCTCD, BILLNUM, CIC, ORIGPVN, or TERMPVN
ExtensionParameter	***ExtensionParameter***s require the CAIN0200 SOC option.
servTranslationScheme	Contains a serving translation scheme
callType	Contains the network call type
satRestriction	Indicates the call should not terminate to a satellite-based trunk
classOfSvc	Contains an index into table MULTICOS for class of service (COS) screening
callBranding	When available, contains an announcement or tone to be played prior to routing
billSequenceNumber	Contains 32 bits of SCP-defined billing data that is stored in the CDR
cainGroup	Contains the group number (field GRPNUM, table CAINGRP) for the CAIN group associated with the trigger when field EXTPARM in table CAINGRP contains CAINGRP.
reorigAllowed	When present, indicates whether reorigination is allowed for the call in progress
univIdx	Contains the universal translation scheme to use for the call (for Signaling System7 [SS7] Global intermachine trunks [IMTs] only)
netinfo	Contains business group information to be sent in the outgoing ISUP IAM

—continued—

Analyze_Route message
NetworkBuilder query (continued)

Table 7-5
Analyze_Route parameters (continued)

Parameter	Definition
callCtrl	Contains a call control value enhancing control over trigger detection point [TDP] evaluation for the call in progress ***Note:*** Ignored if received in a conversational package
primaryTrunkGroupSTS	Contains the serving translation scheme associated with the ***PrimaryTrunkGroup*** routing parameter.
alternateTrunkGroupSTS	Contains the serving translation scheme associated with the ***AlternateTrunkGroup*** routing parameter.
secondAlternateTrunkGroup STS	Contains the serving translation scheme associated with the ***SecondAlternateTrunkGroup*** routing parameter.
overflowRoutingNoSTS	Contains the serving translation scheme associated with the OverflowRoutingNo routing parameter.
accountCode	Contains the account code collected from in-switch translations or by the *O_Feature_Requested* trigger
pinDigits	Contains the collected PIN code, when available, or a PIN collected at *O_Feature_Requested*. It also contains digits collected during conversational digit collection.
billingNumber	Contains a non-standard charge number (for example, CARD, AUTH, ACCT, PIN, or N00).
GenericAddressList	
AlternateOutpulseNo	Contains the digits to be outpulsed when the ***AlternateTrunkGroup*** Number to Outpulse field is 0
SecondAlternateOutpulseNo	Contains the digits to be outpulsed when the ***SecondAlternateTrunkGroup*** Number to Outpulse field is 0
OverflowRoutingNo	Contains an overflow routing number
DialedNoInwardService	Contains a Dialed Number Inward Service (DNIS) value specific to the DNIS specified

—continued—

Analyze_Route message NetworkBuilder query (continued)

Table 7-5
Analyze_Route parameters (continued)

Parameter	Definition
`PortedDialedNo`	Contains the called party address of a successful LNP query
`ForwardCallIndicator`	Indicates that an LNP check has been performed at the SCP
—end—	

Parameter processing

Table 7-6 provides the steps NetworkBuilder call processing uses to process the parameters in an **`Analyze_Route`** message. The presence or absence of these parameters is verified by the switch and appropriate action is taken.

Table 7-6
Parameter processing steps

Step	Call Processing Action
1	Check for billing number. ***Note:*** If the query didn't include a ***`ChargeNumber`*** parameter, and the SCP doesn't return one, a fatal application error is reported.
2	Standard parameters and extension parameters are validated and the appropriate call processing and billing values are updated.
3	When the `classofSvc` extension parameter is provided, multi-COS screening is performed. ***Note 1:*** If multi-COS screening fails, the call is treated (usually COSX). ***Note 2:*** Multi-COS screening may route the call through standard routing.
4	If the `callBranding` extension parameter is present in the SCP response, a tone or announcement is played before routing. ***Note:*** Branding is also performed if a default `callBranding` parameter is datafilled in table CAINXDFT.

Analyze_Route message NetworkBuilder query (continued)

Fatal application errors

Fatal application errors occur when NetworkBuilder call processing is unable to continue due to an unexpected error. Table 7-7 lists errors that can occur after an **`Analyze_Route`** SCP response is returned:

Note: Fatal application errors are handled differently for AXXESS agents and for EDPs. Refer to *UCS DMS-250 CAIN/FlexDial Interactions* or the *UCS DMS-250 NetworkBuilder Application Guide* for more information.

Table 7-7
Analyze_Route fatal application errors

Error type	Definition	Log generated	Reported to SCP?	Error action performed
Unexpected message sequence	The **`Request_Report_BCM_Event`** component is missing from the conversation package.	CAIN200	Yes	ERRACT in trigger table (Note)
Missing conditional parameter	A charge number wasn’t included in the query and one wasn’t returned by the SCP.	CAIN200	Yes	ERRACT in trigger table (Note)
Erroneous data value	SCP sends one or more routing parameters, but CAIN call processing is unable to identify a route index.	CAIN200	Yes	ERRACT in trigger table (Note)

Note: Trigger tables OFFHKIMM, CUSTDP, and SPECDIG provision an error action; AIN final treatment (AINF) is used for all other triggers. The OFFCCODE trigger table defaults to the ROUTE error action.

Nonfatal application errors

Table 7-8 lists errors that can occur while the **`Analyze_Route`** parameters are being processed.

Table 7-8
Analyze_Route nonfatal application errors

Error type	Log generated	Reported to SCP?	Error action performed
Erroneous data value	CAIN100	No	ROUTE
Unexpected parameter	CAIN100	No	ROUTE
Missing conditional parameter	CAIN100	No	ROUTE
SOC for extension parameters (CAIN0200) is idle	CAIN102	No	Extension parameters are ignored
Invalid direct termination parameters	CAIN300	No	Parameters are ignored

Associated logs

CAIN100, CAIN102, CAIN200, CAIN300

Note: For more information on these logs, refer to the *UCS DMS-250 Logs Reference Manual.*

Associated OMs

CAINMSGR, CAINAGOM, CAINTRIG

Note: For more information on these OMs, refer to the *UCS DMS-250 Operational Measurements Reference Manual.*

Continue message

Use

The SCP sends a **`Continue`** message in response to a switch-originated query. This message directs the switch to continue processing the subscription methods at the trigger originating the query and to continue through the call model once all subscription methods are analyzed.

Note: The **`Continue`** message is not expected to contain a **`Request_Report_BCM_Event`** component. If the message contains this component, the transaction is closed and a **`Close`** message with a Close Cause value of `unexpectedCommunication` is sent.

Message parameters

Table 7-9 lists the parameter that may be returned by the SCP:

Table 7-9
SCP Continue message parameters

Parameter	Definition
`AMAAlternateBillingNumber`	Identifies an alternate billing number to which the AIN service should be billed
`AMALineNumber`	Contains digits used to populate one or more of the following CDR fields: ANISP, CLGPTYNO, or PRESIND
`AMAslpID`	Contains a digit string to be populated into the SLPID CDR field
`AMADigitsDialedWC`	Contains digit strings to be populated into one or more of the following CDR fields: PINDIGS, ACCTCD, BILLNUM, CIC, ORIGPVN, or TERMPVN
`ExtensionParameter`	***`ExtensionParameter`***s require the CAIN0200 SOC option.
`billSequenceNumber`	Contains 32 bits of SCP-defined billing data that is stored in the CDR

Continue message (end)

Table 7-9
SCP Continue message parameters

Parameter	Definition
`connectToSCU`	When present, this parameter indicates that CAIN call processing should be terminated and call control should be passed to the programmable service node control unit.
`callCtrl`	Contains a call control value enhancing control over TDP evaluation for the call in progress ***Note:*** `callCtrl` is ignored if received in a conversation package.

Nonfatal application errors

Table 7-10 lists errors that can occur while the **`Continue`** parameters are being processed.

Table 7-10
Continue nonfatal application errors

Error type	Log generated	Reported to SCP?	Error action performed
An AIN 0.2 parameter was sent in an LNP message	CAIN100	No	Parameters are ignored

Associated logs

CAIN102

Associated OMs

CAINMSGR, CAINAGOM, CAINTRIG

Disconnect message

Use

Upon receipt of the **`Disconnect`** message, the switch stops call processing and applies treatment.

Message parameters

Table 7-11 lists the parameters that may be returned by the SCP:

Table 7-11
Disconnect message parameter

Parameter	Definition
`AMAAlternateBillingNumber`	Identifies an alternate billing number to which the AIN service should be billed
`AMALineNumber`	Contains digits used to populate one or more of the following CDR fields: ANISP, CLGPTYNO, or PRESIND
`AMAslpID`	Contains a digit string to be populated into the SLPID CDR field
`AMADigitsDialedWC`	Contains digit strings to be populated into one or more of the following CDR fields: PINDIGS, ACCTCD, BILLNUM, CIC, ORIGPVN, or TERMPVN
`ExtensionParameter`	***`ExtensionParameter`***s require the CAIN0200 SOC option
`billSequenceNumber`	Contains 32 bits of SCP-defined billing data that is stored in the CDR
`treatment`	When present, indicates the treatment to be set as a result of SCP service logic. The SCP must return a valid treatment. Refer to Appendix B, “Treatment codes,” for a list of valid treatments.
—end—	

Fatal application errors

Fatal application errors occur when NetworkBuilder call processing is unable to continue due to unexpected errors.

Nonfatal application errors

Table 7-12 lists errors that can occur while the **`Disconnect`** parameters are being processed.

Table 7-12
Disconnect nonfatal application errors

Error type	Log generated	Reported to SCP?	Error action performed
SOC for extension parameters (CAIN0200) is idle	CAIN102	No	Extension parameters are ignored

Associated logs

The switch generates a TRK138 log when the treatment identified in the `treatment` parameter is applied to the call.

Associated OMs

CAINMSGR, CAINAGOM, CAINTRIG

Send_To_Resource and Connect_To_Resource messages (end)

Use

In response to a NetworkBuilder query, the **`Send_To_Resource`** and **`Connect_To_Resource`** messages instructs the switch to perform one of the following:

- play an announcement and disconnect
- play one or more announcements and collect digits
- route to an intelligent peripheral

In response to an LNP query, if the **`Send_To_Resource`** or **`Connect_To_Resource`** message is received in a Conversation package, the service switching point (SSP) sends a **`Resource_Clear`** or **`CTR_Clear`** message in a Response package with a ***`ClearCause`*** value of `taskRefused`. If a **`Send_To_Resource`** or **`Connect_To_Resource`** message is received in a Response package with a disconnect flag, the SSP attempts to perform default routing using the called party address.

For more information regarding these messages, refer to the *UCS DMS-250 NetworkBuilder Application Guide*.

Limitations and restrictions

None

Associated logs

None

Associated OMs

CAINMSGR, CAINAGOM, CAINTRIG

NetworkBuilder tools

NetworkBuilder offers several tools for troubleshooting and tracking purposes.

The following tools are available:

- CAINTEST
- Service control point (SCP) simulator
- Software optionality control (SOC)

  Note: Although the SOC command set is not a NetworkBuilder tool, it is included here because it is required to enable NetworkBuilder.

- Translation verification (TRAVER)

  Note: AXXESS agents use the FLEXSIM tool rather than TRAVER. Refer to *UCS DMS-250 CAIN/FlexDial Interactions* for more information.

CAINTEST

CAINTEST is the NetworkBuilder message query test tool. You can create and send test Transaction Capabilities Application Part (TCAP) queries to the SCP. The SCP responds as if an actual call were being made and does not recognize any difference between a CAINTEST-generated message and a call-generated message.

CAINTEST provides the ability to do the following:

- generate queries for testing SCP interaction
- display SCP or SCP simulator responses
- manually enter into conversation with the SCP
- set and clear message parameters
- list parameters for any particular switch-generated message
- override the CAIN_T1_TIMEOUT parameter for a particular test query
- display command syntax

- support the maintenance and verification of the SCP or SCP simulator

CAINTEST supports the following items:

- ***UserID*** parameters values DN and TRK
- global title value CAIN_OFCD_GT
- ***TriggerCriteriaType*** values
- receipt of the ***Analyze_Route*** parameter, FCI
- receipt of the GAP ported address type value
- ***ChargePartyStationType*** parameter
- lnpReceived extension parameter
- ***JurisdictionInfo*** parameter

For CAINTEST command information, refer to *UCS DMS-250 NetworkBuilder Application Guide*.

SCP simulator

NetworkBuilder includes an SCP simulator used to test SCP responses to NetworkBuilder queries. Simulator logic is provided through user-defined datafill.

You can enter datafill for the simulator to perform NetworkBuilder testing, as well as robustness testing. Through datafill, the simulator provides the following SCP services:

- accept and decode TCAP messages
- detect and report errors
- check incoming message parameters against a customer-defined database
- determine appropriate response messages, including error cases
- encode response message into TCAP message format
- send TCAP messages to NetworkBuilder
- conversation

For more information regarding this tool, refer to the *UCS DMS-250 NetworkBuilder Application Guide*.

SOC

Nortel Networks uses SOC to define and deliver software in product computing module loads (PCL). Nortel Networks categorizes all functionality in a PCL as either base or optional. Base functionality is available for immediate use. Optional functionality is grouped into commercial units called SOC options, which can be purchased by operating companies. SOC options correspond to functional groups and functions and are controlled by Nortel Networks-supplied passwords.

SOC is the tool for managing options in a PCL. These options reside in the software. When an operating company purchases an option, SOC allows the company to monitor and control its use. You can order, activate, and use these options without a software reload or restart.

LNP SOC option CAIN0700 provides the switch with the ability to do the following:

- perform OFFCCODE trigger table lookup
- set bit M in the outgoing FCI parameter based on the SCP response from a NetworkBuilder query.

This SOC will not interfere with the ability of the switch to do the following:

- pass-through an FCI and/or GAP parameter
- convert an GAP to the called party address when terminating to a non-ISUP trunk
- handle ISUP release cause 26 according to existing switch logic
- place an LRN in the CALLEDNO field of the CDR when incoming LNP information is received.

For SOC command information, refer to the *UCS DMS-250 Software Optionality Control (SOC) User's Manual*.

TRAVER

The TRAVER (translation verification) tool simulates a call from a user specified originating trunk to a user-specified address. TRAVER examines and displays translation and routing data for a single call leg.

TRAVER performs the following:

- verifies the translation tables
- aids in debugging and analyzing translation and routing datafill
- helps determine reasons for unexpected results and changes required to achieve the expected results

TRAVER is capable of displaying the following:

- tables used to translate and route calls
- treatment

- NetworkBuilder subscription method and group
- tuple (from the appropriate trigger table) where trigger criteria was met
- limited messaging parameters

TRAVER functionality has been broadened to include queries through the OFFCCODE trigger table.

For TRAVER command information, refer to the *UCS DMS-250 Commands Reference Manual*.

FLEXSIM

The FLEXDIAL tool FLEXSIM has been updated to provide the capability to generate an SS7 IAM message with bit M of the FCI set and a GAP with an address type of PORTEDNO.

For more information regarding this tool, refer to *UCS DMS-250 CAIN/FlexDial Interactions*.

CDRTST

The CDRTST tool includes the LNPCHECK and PORTEDNO fields in the test CDRs that it generates.

FCDRSRCH

This feature enables the FCDRSRCH tool to search billing files for CDRs containing the LNPCHECK and PORTEDNO fields.

List of terms

ACG
automatic code gapping

advanced intelligent network
A network that allows the switch to off-load some of the call processing functions to an intelligent service control point (SCP).

AIN
advanced intelligent network

AIND
AIN disconnect treatment

AINF
AIN final treatment

ANI
automatic number identification

ANISCUSP
Automatic Number Identification Screening Customer Profile table. May be used to assign a CAIN group to a particular ANI.

AUTHCDU2
Authcode Database 2 table. May be used to assign a CAIN group to a particular authorization code.

AUTHCDU3
Authcode Database 3 table. May be used to assign a CAIN group to a particular authorization code.

AUTHCDU4
Authcode Database 4 table. May be used to assign a CAIN group to a particular authorization code.

AUTHCDU5
: Authcode Database 5 table. May be used to assign a CAIN group to a particular authorization code.

AUTHCODU
: Authcode Database table. May be used to assign a CAIN group to a particular authorization code.

authorization code
: A unique multidigit code that identifies an authorized subscriber. Authorization codes are usually 5-7 digits (due to UCS DMS-250 limitations) and identify a subscriber, bill a call, prevent unauthorized network use, determine the originating caller's class of service, and control access to special features. For example, a caller can be required to enter an authorization code to retrieve voice mail messages.

automatic number identification
: billing number for the calling party provided to the IEC from the LEC

BC
: bearer capability

bearer capability
: A characteristic associated with a directory number to indicate the type of call (voice or data) and the rate of transmission allowed.

C7GTT
: CCS7 Global Title Translation table. Maps a translation type (defined in table C7GTTYPE) to a CCS7 network address.

C7GTTYPE
: CCS7 Global Title Type table. Maps a CCS7-defined translation to a network-defined global title translation type.

C7LINK
: CCS7 Link table. Makes the association between the physical equipment of the link and the logical view of the link as a member of a linkset.

C7LKSET
: CCS7 Link Set table. Defines the characteristics of a linkset. A linkset is a set of links used as a group. Each link carries traffic between the origination point code and a destination point code. The table also defines attributes that are common to all links in the link set. The links are defined in table C7LINK.

C7LOCSSN

CCS7 Location Subsystem Number table. Defines the subsystems located on the switch.

C7NETSSN

CCS7 Network Subsystem Number table. Provides the set of remote point codes (PC) and subsystems, at the remote PCs, where messages are routed by the SCCP. A PC is a node in the CCS7 network that may be an SSP, an STP, or an SCP.

C7NETWRK

CCS7 Network table. Describes the signaling networks in use in a switching office.

C7RPLSSN

CCS7 Replicate Subsystem table. Provides the set of remote subsystem replicate pairs. It has a one part key, the subsystem name. For each subsystem a list of PC pairs at which the replicated subsystems reside must be given.

C7RSSCRN

CCS7 Remote Subsystem Concerned Node table. Provides a list of concerned nodes for a remote subsystem point code combination. The table has a two part key. The first part is the PC and the second part is the subsystem name. The PC and subsystem combination must be datafilled in table C7NETSSN.

C7RTESET

CCS7 Route Set table. Associates linksets used as possible routes for each signaling point in the network. An office point code identifies a signaling point within any network. Each office point code must have a routeset. The information in this table records which routes and linksets can carry the signaling information to the destination signaling point. This table is also used for alternate routing decisions.

CAIN routing parameters

Parameters provided in an **`Analyze_Route`** response. The parameters are: ***`PrimaryTrunkGroup`***, ***`AlternateTrunkGroup`***, ***`SecondAlternateTrunkGroup`***, ***`CalledPartyID`***, and the ***`GenericAddressList`***'s `OverflowRoutingNo`.

CAINCONV

CAIN Conversation table. Controls SCP simulator interaction during conversation with the switch.

CAINGRP

CAIN Group table. Identifies CAIN groups and trigger sets used for NetworkBuilder subscription.

CAINKEY

CAIN Key table. Determines a range of possible responses for a defined three-part key into the SCP simulator. The range of possible responses is represented by an option vector of indexes into table CAINMTCH.

CAINMTCH

CAIN Matching table. Screens and selects possible responses for the SCP simulator

CAINPARM

CAIN Parameters table. Assigns NetworkBuilder office parameter values.

CAINRESP

CAIN Response table. Contains response data to return to the switch. The SCP simulator's encoder takes this data and builds a Transaction Capabilities Application Part (TCAP) message.

CAINRSRC

CAIN Resource table. Maps the data returned from the SCP to a resource available on the switch.

CAINUID

CAIN User Identification table. Provides symbolic names for trunk groups and switches used in the simulator. It is similar to table CLLI in function. The use of symbolic names rather than numbers provides enhanced clarity when datafilling the simulator tables.

CAINXDFT

CAIN Extension Parameter Defaults table. Defines default values for five extension parameters: `servTranslationScheme`, `callType`, `satRestriction`, `classofSvc`, and `callBranding`.

call branding

Tones or announcements (returned from the SCP or provisioned in table CAINXDFT) played by the switch as directed by NetworkBuilder call processing before routing is attempted.

call detail record

Formatted billing data used to generate subscriber billing.

call model

Generic representation of a basic call, in terms of the processing activities required to establish, maintain, and clear a call.

call processing

The function of call processing software is to establish connections among telephony agents. A number of functional steps are required to process a call, such as detecting the incoming call, receiving digits, analyzing (translating) digits to determine call destination, selecting terminating agent, establishing connection, signaling to and detecting an answer from the terminating agent, and detecting disconnect.

CALLATTR

Call Attributes table. Used to provision a PRI call attribute.

called number

The number of the party receiving the call. Also known as called party ID.

called party

The end user that receives a call.

calling line identification

In data transmission, a feature provided by the network that allows a called terminal to be notified by the network of the address from which the call was originated.

calling number

The number of the party initiating the call. This number can identify the origin of a call to the called party. Also known as calling party number or calling party ID.

calling party

The end user that originates a call.

carrier identification code

Three- to four-digit number that identifies which interexchange carrier a call will use. Subscribers can dial these digits with each long distance call, or can pre-subscribe to a particular carrier and let the digital switch software add the CIC.

CCS7

Common Channeling Signaling 7. A digital, message-based network signaling standard defined by the CCITT which separates call signaling information from voice channels so that interoffice signaling is exchanged over a separate signaling link.

CDR

call detail record

CIC

Carrier Identification Code

CLI
calling line identification

CLLI
common language location identifier

CM
computing module

common language location identifier
A standard identification method for trunk groups, tones, and announcements.

computing module
The processor and memory of the dual-plane combined core used by the DMS SuperNode. Each CM consists of a pair of central processing units (CPU) with associated memory that operate in a synchronous matched mode on two separate planes. Only one plane is active; it maintains overall control of the system while the other plane is on standby.

conversation package
A transaction capabilities application part (TCAP) package expecting a reply. It can be sent by the service control point (SCP), or adjunct, to the service switching point (SSP) that instructs the SSP to perform a user interaction, collect digits, and send a reply to the SCP. The SSP can also send a conversation package to the SCP.

COS
class of service

CPA
Called Party Number. Contains the number to use in translations and routing; also identifies the Nature of Address (NOA) and Numbering Plan, for LNP calls this parameter can contain the Location Routing Number (LRN).

CPID
calling party identification

CUSTDP
Customized Dialing Plan table. Defines trigger criteria for *Customized_Dialing_Plan*.

DAL
dedicated access line

data schema

Tables that direct how the NetworkBuilder calls are processed.

dedicated access line

Network connection, often leased from a local exchange carrier or competitive access provider, that provides a direct link from a customer to the long distance network. Typical DALs include outbound WATS lines, PBX tie trunks, and foreign exchange lines.

digital recorded announcement machine (DRAM)

A peripheral module (PM), developed for the DMS switch, in which voice messages are stored in digital form, providing access to up to 30 different service voice announcements.

Donor Switch

The switch from which a number was ported. The switch which used to service the end user before Service Provider Portability.

dual-tone multifrequency (DTMF) signaling

A signaling method employing set combinations of two specific voice-band frequencies, one of which is selected from a group of four low frequencies, and the other from a group of three or four relatively high frequencies.

EANT

equal access network trunking

fast interdigit (FIDT) timer

time allowed for the subscriber to dial digits between the minimum and maximum required

FCC

Federal Communications Commission

FCI

Forward Call Indicator. Used to identify special call capabilities and features such as the LNP check.

Federal Communications Commission

An agency of the U. S. Government that regulates standards and companies within all aspects of the communications industry (radio, television, telephony).

FGD

Feature Group D

FIDT

fast interdigit timer

FlexDial

FlexDial Framework is a UCS DMS-250 feature that allows you to program the call origination side of a call to implement customized functionality. UCS06 software does not support CAIN/FlexDial interaction.

GAP

Generic Address Parameter. Used to relay different types of address digits. Identifies the associated Nature of Address (NOA) and Numbering Plan. Used by LNP to store the Called Party Address.

Note 1: More than one GAP parameter can be present in an IAM.

Note 2: When LNP uses the GAP parameter to house the Called Party Address, the Type of Address value is set to Ported Dialed Number.

GDP

generic digits parameter

generic digits parameter

ISUP parameter used to transport generic digits with a specified identifier tag

global title

The application's address (translated to a PC + SSN). The SCCP global title translation (GTT) function is required to translate a GT into a valid network address.

global title translation

The process that translates an application-specific address (such as a dialed 800 number) into the Common Channel Signaling No. 7 (CCS7) PC subsystem address, usually that of the appropriate service control point (SCP).

GT

global title

GTT

global title translation

Home LRN

home location routing number. A unique LRN assigned to a switch.

hotline

A connection that has another address mapped to it. This second address can be filed in a table, and therefore not changeable by the end user, or it can be operated by an authcode for an effect similar to speed dialing.

IAM

Initial Address Message

IN

intelligent network

IN/1

UCS proprietary intelligent network protocol for offboard databases with limited call control

IEC

interexchange carrier

IMT

intermachine trunk

information digit

Digit received by the switch either in the incoming digit stream or with the incoming signaling information. An information digit carries additional information about the call that cannot be determined by the address digits. The type of information digit received and the method it is received depends on the incoming signaling system.

Integrated Services Digital Network

A network that provides end-to-end digital connectivity using CCS7 to support a wide range of voice and data services to the end-user.

intelligent peripheral

Contains functionality and resources for exchanging information (such as voice announcements and dual-tone multifrequency digit collection) with a subscriber.

intermachine trunk

A trunk that connects the UCS DMS-250 IEC networks.

IP

intelligent peripheral

ISDN

Integrated Services Digital Network. A network that provides end-to-end digital connectivity using CCS7 to support a wide range of voice and data services to the end-user.

ISDN User Part

SS7 protocol that defines the messages, parameters, and procedures to set up and tear down all circuit switched calls, both ISDN and non-ISDN, in U.S. SS7 networks. It includes support for ISDN Supplementary Voice services and interworks with Q.931/932 to provide end-to-end ISDN.

ISUP

ISDN User Part. The signalling portion of SS7 required to provide voice and non-voice service in ISDN networks.

JIP

Jurisdiction Information Parameter. Contains geographic and service provider information that is associated with the calling party. For LNP this can be interpreted as the LRN of the originating switch.

LEC

local exchange carrier

link set

Two or more redundant links between two nodes.

LNP

local number portability

LNP SCP

LNP service control point

logs

Records of call activities that occur within the switch.

LRN method

Location Routing Number method. LRN method is a technique used to query an SCP (Service Control Point) when many numbers have been ported in an end-office or when it is undesirable to route to the switch from which the number was ported (“donor” switch), receives an SCP response message, and routes the call based on the information returned.

LRN

location routing number. The LRN is a 10-digit number (DN) that uniquely identifies an end-office to the network for call routing purposes.

MCCS

mechanized calling card service

mechanized calling card service

The service that allows a call to be billed to a calling card number.

message parameters

Parameters within a message. Each message type has its own set of mandatory and optional parameters.

message set

the set of standard messages required by an advanced intelligent network (AIN) specification, such as Bellcore’s Release 0.2, for communication among AIN network elements.

message switch and buffer

A peripheral module used by the switch, along with a signaling terminal, to act as an interface to and operate within a common channel signaling environment. The message switch and buffer supports the signaling terminal and routes the messages received by the signaling terminal through the network module to the digital trunk controller. The message switch and buffer also receives messages sent from central control and routes them to the signaling link through the signaling terminal. A different configuration of the message switch and buffer exists for each of the two protocols used to implement common channel signaling.

message switch and buffer 7

The message switch and buffer (MSB) for Common Channel Signaling 7 (CCS7) protocol. *See also* message switch and buffer (MSB)

MSB

message switch and buffer

MSB7

message switch and buffer 7

MTP

Message Transfer Part.

NETBUSY

Network Busy table. Defines trigger criteria for *Network_Busy*.

numbering plan area

Any of the designated geographical divisions of the United States, Canada, Bermuda, and Northwestern Mexico within which no two telephones have the same seven-digit number. Each NPA is assigned a unique three digit area code for World Zone 1 dialing.

OCLDBUSY

O Called Party Busy table. Defines trigger criteria for *O_Called_Party_Busy*.

OFFHKIMM

Off Hook Immediate table. Defines trigger criteria for *Off_Hook_Immediate*.

off-hook

The condition existing in telephone operations when the receiver or handset is removed from its hookswitch.

OFTRREQ

O Feature Requested table. Defines trigger criteria for *O_Feature_Requested*.

OM

operational measurements. Measurements of data are collected and displayed as the switch performs various operations.

ONOANSWR

O No Answer table. Defines trigger criteria for O_No_Answer.

operational measurements

The software resources of the switch that control the collection and display of measurements taken on an operating system. The OM subsystem organizes the measurements data and manages its transfer to displays and records. The OM data is used for maintenance, traffic, accounting, and provisioning decisions.

originating switch

The switch from which a call originates.

PANI

pseudo-automatic number identification

parameter

Data contained within a message.

partial dial (PDIL) timer

time allowed between each subscriber dialed digit (before minimum number of digits are dialed)

PBX

private branch exchange

PC

point code. Address of a node in the CCS7 network.

PCL

product computing module loads

PDIL

partial dial timer

per-trunk signalling

A conventional telephony method of signaling that multiplexes the control signal of a call with voice or data over the same trunk (in-band signaling).

permanent signal timer

time allowed before the subscriber enters the first digit

personal identification number

Authorization number (usually composed of the caller's telephone number plus a four-digit code) that allows subscribers to access their long distance carriers when away from home.

PIC

point in call. Represents the call processing functionality required by a basic two-party call. A generic representation of a sequence of switch-based call processing actions considered essential to establish, maintain, or clear a two-party call. PICs are separated by trigger detection points (TDPs).

PIN

personal identification number

point in call

PRI

primary rate interface

PRIBCHNL

PRI B-CHANNEL table. Defines trigger criteria for *PRI_B-Channel*.

primary rate interface

An interface that carries nB+D channels over a digital DS-1 facility (23B+D in North America and 30B+D in Europe). PRI is used to link private networking facilities, such as private branch exchanges (PBX), local area networks (LAN), and host computers with a standardized architecture acting as the bridge between private switching equipment and the public network. Formerly known as primary rate access.

private branch exchange

A private telephone exchange, either automatic or attendant operated, serving extensions in an organization and providing access to the public network.

pseudo automatic number identification

A 10-digit translations code derived from a combination of the authorization code, personal identification number, and serving number plan area number.

PSIG
: permanent signal timer

PSN
: programmable service node

PTS
: per trunk signaling

query
: A type of communication message sent by the service switching point (SSP) to the service control point (SCP), or adjunct, requesting call processing instructions. In AIN Release 0.2, the message is contained in a transaction capabilities application part (TCAP) query package.

Recipient Switch
: The switch to which a number has been ported. The switch which now services the end user.

REL
: CCS7 release message

response package
: A transaction capabilities application part (TCAP) package containing one or more messages sent in response to another TCAP package. The service control point (SCP), or adjunct, can use a response package to instruct the service switching point (SSP) to perform an activity. The SSP can also send response packages.

response processing
: The service switching point (SSP) receives and processes response messages from the service control point (SCP), or adjunct. There are several different types of response messages, each of which is handled differently according to the information it contains.

route
: Signaling path from one node to another (may go through multiple nodes).

route set
: Set of all routes from one node to another.

SCCP
: Signaling Connection Control Part. Software protocol acting as an interface between OSI layers.

SCE
: service creation environment

SCP

service control point A node in a Common Channel Signaling No 7 (CCS7) signaling network that supports application databases. The function of an SCP is to accept a query for information, retrieve the requested information from one of its application databases, and to send a response message to the originator of the request.

service control point

A node in a Common Channel Signaling No 7 (CCS7) signaling network that supports application databases. The function of an SCP is to accept a query for information, retrieve the requested information from one of its application databases, and to send a response message to the originator of the request.

service switching point

A switch that is capable of interacting with the Common Channel Signaling No. 7 (CCS7) network databases. It contains hardware to support CCS7 signaling, software to control call processing and also create network database query messages, and software to interpret network database response messages.

serving translation scheme

The scheme the UCS DMS-250 switch uses to translate and route a call. STS codes are three digits codes (000-999). The switch uses the three digits serving translation scheme codes (000-999) to derive routing information.

signal transfer point

A switch that is used to provide signaling link connections between switches. That is, it is a tandem node for Common Channel Signaling No. 7 (CCS7) signaling links and contains hardware to support CCS7 hardware and software to route CCS7 messages. It does not contain any software to create or interpret CCS7 messages. STPs are deployed in pairs. If one STP fails, the mate takes over, ensuring that service continues without interruption. One of the STPs primary functions is performing global title translations.

signaling

Communication between switches, or switches and end points, to set-up, manage, and tear-down calls. Signaling methods include dial pulse (rotary dial), dual-tone multifrequency (DTMF) (touch-tone), and digital “packet” technology (ISDN, SS7)

Signaling Connection Control Part

A level of common Channel Signaling No. 7 (CCS7) layered protocol. The main functions of the SCCP include the transfer of signaling units with or without the use of a logical signaling connection at the provisioning of flexible translations (GTT) for different applications.

signaling link
The communication channel between two nodes.

SIOTRK
Shared Interoffice Trunk table. Defines trigger criteria for *Shared_Interoffice_Trunk*.

SMS
service management system

SOC
software optionality control

SPECDIG
Specific Digit String table. Defines trigger criteria for *Specific_Digit_String*.

SSN
subsystem number. Address of a subsystem (application) at a node in the CCS7 network (for example. CAINTEST).

SS7
Signaling System 7. An ANSI standard protocol used by networks.

SSP
service switching point (DMS–250)

STP
signal transfer point

STR-Connection
A connection that is made from an SSP to an IP over an ISDN IP interface in response to a **`Send_To_Resource`** message.

SUS
CCS7 suspend message

TANDMRTE
Tandem Routing table. Provisions routing through tandem switches within the IEC network to reach the required terminating switch.

TCAP

Transaction Capabilities Application Part. A service that provides a common protocol for remote operations across the Common Channel Signaling No. 7 (CCS7) network. The protocol consists of message formatting, content rules, and exchange procedures. TCAP provides the ability for the service switching point (SSP). TCAP is used by the integrated services digital network (ISDN) layer facility message to transport service information for transaction signaling, not associated with an active call, over primary rate interface links.

TDP

trigger detection point. Point within the call model where the switch can temporarily suspend call processing and send a message to the SCP.

TERMRTE

Termination Routing table. Provisions routing to a terminating network directly connected to the current network.

Transaction Capabilities Application Part

A service that provides a common protocol for remote operations across the Common Channel Signaling No. 7 (CCS7) network. The protocol consists of message formatting, content rules, and exchange procedures.

translation verification

A diagnostic tool that allows the operating company to access and simulate a telephone call in software and display the tables and tuples used to establish the lines, trunks, or positions to which a call is routed.

TRAVER

translation verification

treatment

The method by which a call is disconnected or ended.

TRID

transaction identifiers

trigger

A trigger defines the actions taken once trigger criteria is met at a TDP on the SSP.

trigger criteria

Trigger criteria defines the call conditions that must be met for a particular call to trigger. A trigger may contain criteria of one or more trigger criteria types. In order for a call to trigger, all trigger criteria must be met.

trigger detection point

A point in basic call processing, as modeled by the basic call model (BCM), which identifies when a service control point (SCP) can receive notification of a given event and influence subsequent call processing. TDPs are located at transitions between points in call (PIC).

trigger tables

Trigger tables store information about the advanced intelligent network (AIN) application, triggers, trigger detection points, trigger criteria, transport protocol.

triggering

The process where a call indicates that it requires advanced intelligent network (AIN) service(s).

TRKGRP

Trunk Group table. Assigns a CAIN group to a particular agent.

VAMP

variable AIN messaging platform

VAMPTRID

VAMP Transaction Identifiers table. Provisions the key resources used in Carrier AIN messaging, including transaction and component identifiers and message buffers.

VPN

virtual private network

Ordering information

Use the following table for ordering Nortel Networks NTPs (Northern Telecom Publications) and Product Computing-Module Loads (PCLs):

Type of product	Source	Phone	Cost
Technical documents (paper or CD-ROM)	Nortel Networks Product Documentation	1-877-662-5669, Option 4 + 1	Yes
Individual NTPs (paper)	Merchandising Order Service	1-800-347-4850	Yes
Marketing documents	Sales and Marketing Information Center (SMIC)	1-800-4NORTEL (1-800-466-7835)	No
PCL software	Nortel Networks	Consult your Nortel Networks sales representative	Yes

When ordering publications on CD

Please have the CD number and software version available, for example, **HLM-2621-001 02.02**.

When ordering individual paper documents

Please have the document number and name available, for example, **297-2621-001, UCS DMS-250 Master Index of Publications**.

When ordering software

Please have the eight-digit ordering code, for example, **UCSE0012**, as well as the ordering codes for the features you wish to purchase. Contact your Nortel Networks representative for assistance.

Digital Switching Systems

UCS DMS-250

Local Number Portability (LNP)

Application Guide

Product Documentation–Dept 3423
Nortel Networks
P.O. Box 13010
RTP, NC 27709–3010
1-877-662-5669, Option 4 + 1

Publication number: 297-2621-371
Product release: UCS12
Document release: Standard 05.02
Date: November 1999
Printed in the United States of America

NORTEL NETWORKS™

How the world shares ideas.