



CHAPTER 3

Local Exchange Routing Guide

Revised: April 22, 2009, OL-8720-10

Introduction

This chapter provides a basic understanding of the Cisco BTS 10200 Softswitch local exchange routing guide (LERG). This chapter is divided into the following sections:

- [Local Exchange Routing Guide Support](#)
- [LERG, TNS, and Additional SIP Extensions for CMS-MGC Separation](#)

Local Exchange Routing Guide Support

Local Exchange Routing Guide Support enables you to utilize the routing data from the LERG files, and it allows you to view, add, delete, and change the data stored in the LERG tables. Also, it provides the means to schedule future LERG updates. A new set of commands and a script have been added to provision the LERG data. More information on these commands and on the mechanism for provisioning the LERG data are presented in the subsequent sections.

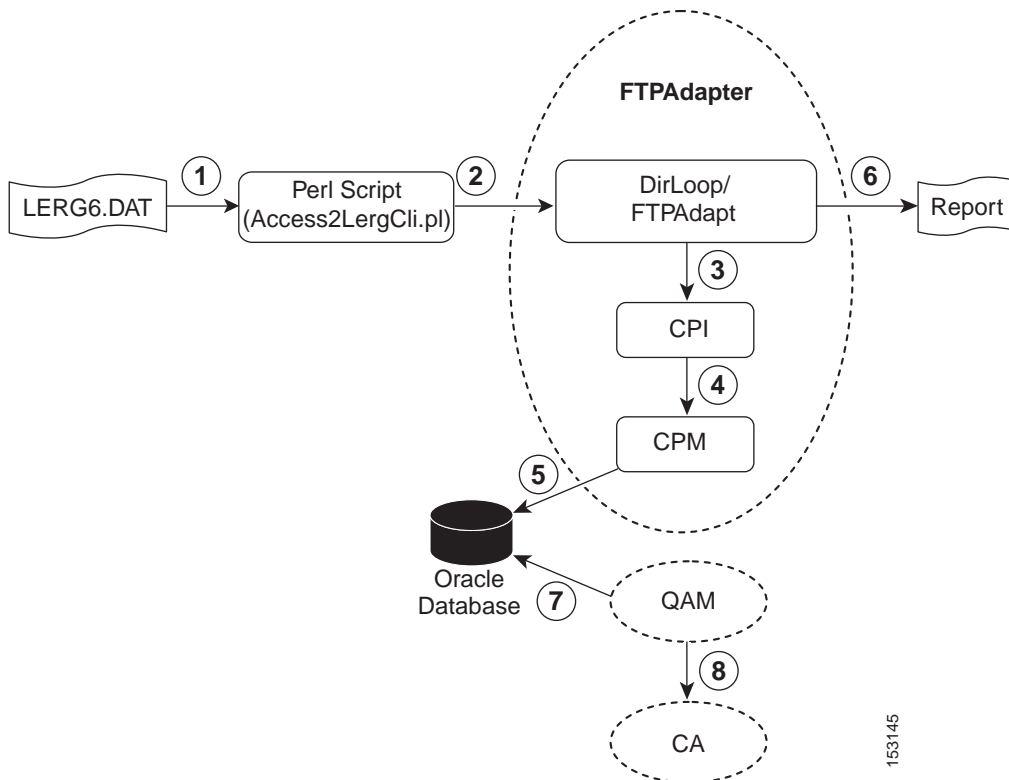
The BTS 10200 operations and maintenance (OAM) application allows users to configure the LERG data from the LERG6INS.DAT and LERG6.DAT files. The LERG6.DAT contains the data which needs to be configured on the BTS 10200 once at any given point in time. The LERG6INS.DAT contains the data which needs to be configured on a specific date in the future. The following two mechanisms are provided on the BTS 10200 OAM application:

- A single command to generate BTS 10200-specific LERG commands from the input LERG6.DAT. This command parses the LERG6.DAT and creates a file containing the BTS 10200-specific command list. This command file is then loaded by the means of the ftpAdapter and applied to a corresponding table in the BTS 10200 OAMP application. The command file is also sent to the Call Agent.
- A single command to generate the BTS 10200 specific LERG commands from the input LERG6INS.DAT file. This command parses the LERG6INS.DAT file and creates a file containing the BTS 10200-specific command list with appropriate start times. This command file is then loaded by means of the ftpAdapter and applied to the corresponding table in the BTS 10200 OAMP application. The command file is also sent to the Call Agent at the specified start time.

LERG6 Update

This section describes the LERG6 file update process. Refer to [Figure 3-1](#) while reviewing the step-by-step LERG6 file update process.

Figure 3-1 LERG6 Update Process Flow



- Step 1** The first step in the LERG6 update involves the creation of a BTS 10200-specific command file using the Access2LergCli script, which is available in the /opt/ems/bin directory. The LERG6.DAT file is provided as an argument to the **lergToCLI** command along with the output file name. The syntax of the **lergToCLI** command is `lergToCLI <<LERG INPUT DATA>> <<OUTPUT FILE>>`.

Example:

The command string `/opt/ems/bin/lergToCLI LERG6.DAT /opt/ems/ftp/deposit/lergCmds` generates the output file lergCmds in the /opt/ems/ftp/deposit directory.

- Step 2** The output file generated by the **lergToCLI** command contains the BTS 10200-specific command line interface (CLI) commands. The output file is copied to the /opt/ems/ftp/deposit directory.



Note

Verify that the generated file has the expected owner. If the file does not have the expected owner, change the owner, using the unix **chown** command.

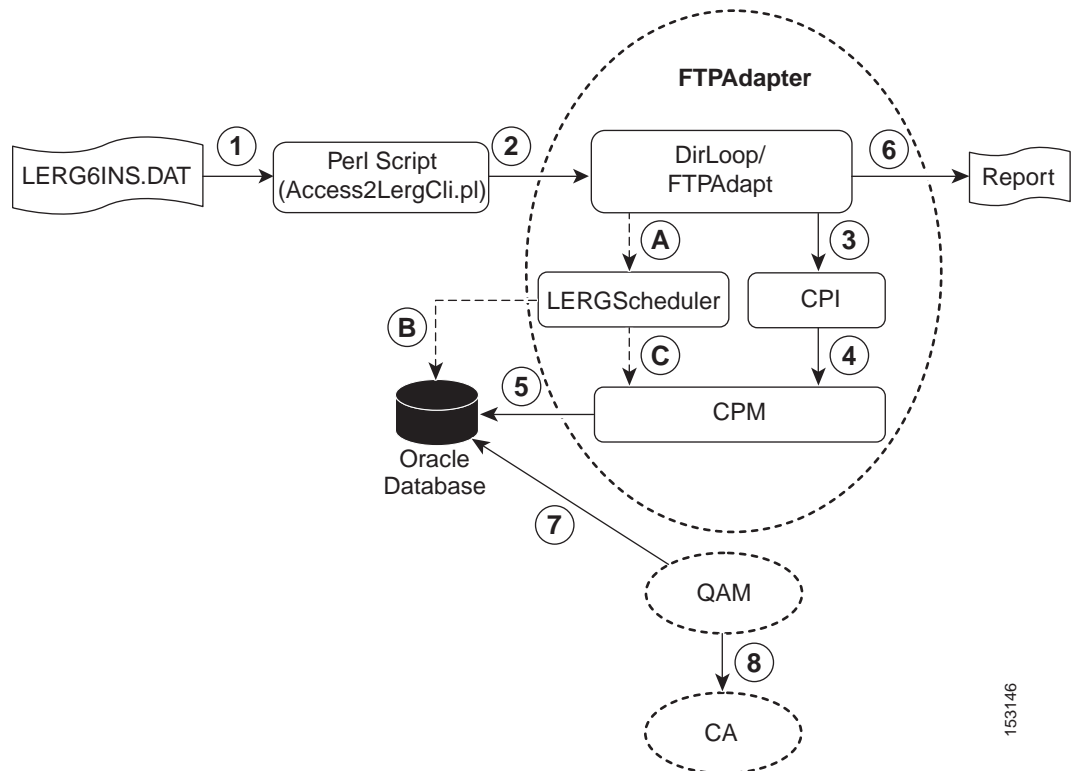
- Step 3** The DirLoop thread, which is polling the deposit directory, reads the CLI commands contained in the output file.

- Step 4** The CLI commands are then parsed and validated by the Command Processing Infrastructure (CPI) module.
- Step 5** The CLI commands are then added into the Oracle database by the Request Manager component of the Command Processing Module (CPM). The Request Manager also creates a transaction request statement and makes an entry into the Transaction Queue.
- Step 6** The result of the CLI command execution is contained in a transaction report.
- Step 7** The Queuing and Audit Manager (QAM) process reads the transaction request statement from the Transaction Queue.
- Step 8** The QAM then forwards the transaction request statement to the Call Agent. Upon receiving the reply from the Call Agent, the OAM deletes the corresponding transaction request statement from the transaction queue.

LERG6INS Update

This section describes the LERG6INS file update process. Refer to [Figure 3-2](#) while reviewing the step-by-step LERG6INS file update process.

Figure 3-2 LERG6INS Update Process Flow



153146

- Step 1** The first step in the LERG6INS update involves the creation of a BTS 10200-specific command file using the Access2LergCli script, which is available in the /opt/ems/bin directory. The LERG6INS.DAT file is provided as an argument to the **lerg6ToCLI** command along with the output file name. The syntax of the **lergToCLI** command is *lergToCLI <<LERG INPUT DATA>> <<OUTPUT FILE>>*.

Example:

The command string */opt/ems/bin/lergToCLI LERG6INS.DAT /opt/ems/ftp/deposit/lergCmds* will generate the output file *lergCmds* in the /opt/ems/ftp/deposit directory.

- Step 2** The output file generated by **lergToCLI** command will contain the BTS 10200 specific CLI commands. The output file is copied to the /opt/ems/ftp/deposit directory.

The Access2LergCli perl script intelligently parses contents of the LERG6INS file and prepares a list which includes parameters, noun, verb and start time. The start time is the day on which the LERG data needs to be applied to the BTS 10200.



Note Verify that the generated file has the expected owner. If the file does not have the expected owner, change the owner using the unix **chown** command.

- Step 3** The DirLoop thread, which is polling the deposit directory, reads the CLI commands contained in the output file.
- Step 4** The CLI commands are then parsed and validated by the CPI module.
- Step 5** The Request Manager component of CPM distinguishes the regular LERG6 update from scheduled LERG6INS based on the noun. It creates a scheduled job and adds it into SCHEDULED_COMMAND and SCHEDULED_PARAM_TABLE.
- Step 6** The result of the CLI command execution is contained in a transaction report.
- During the start up of the File Transfer Protocol (FTP) Adapter (DLP process), a thread called LERG Scheduler is created to handle the periodic LERG6INS updates.
 - This thread wakes up once a day to check the SCHEDULED_COMMAND table for any pending LERG-specific jobs.

You can configure the duration of this check by changing the LERGDURATION parameter in *bts.properties* file. We recommend that users not change this parameter. Improper tuning of this parameter might affect the performance of the DLP process.



Note Modification of the *bts.properties* file should not be attempted without Cisco TAC support or supervision.

- If the LERG Scheduler finds any LERG-related jobs in the SCHEDULED_COMMAND, it creates a corresponding LERG6 request and hands it over to CPM. The Request Manager component of CPM updates the LERG6 table in Oracle and creates an entry in the Transaction Queue.
- Step 7** The QAM process reads the transaction request statement from the Transaction Queue.
- Step 8** The QAM then forwards the transaction request statement to the Call Agent. Upon receiving the reply from the Call Agent, OAM deletes the corresponding transaction from the transaction queue.

LERG6INS Time Schedule

The LERG6INS scheduler is part of the FTP adapter process (DLP process). This scheduler is driven by the java timer and timer class architecture. The scheduler schedules the task for a specified duration. This duration is specified in the bts.properties file.



Note

We recommend that you not change the duration specified in the bts.properties file.



Note

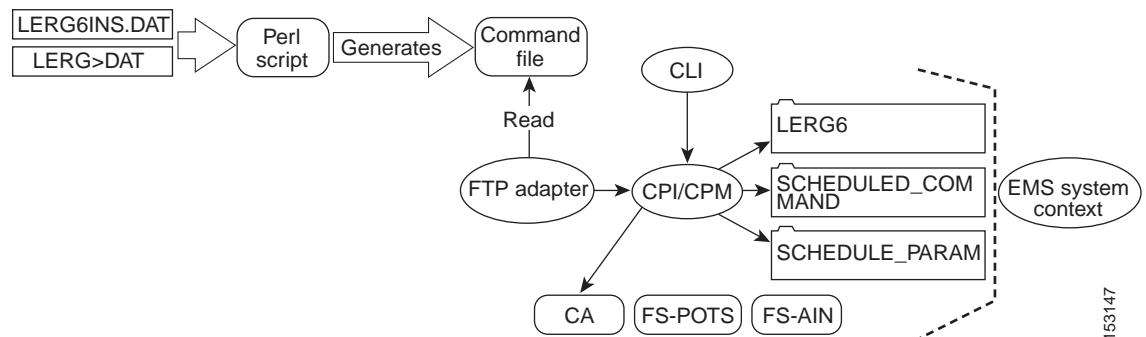
Modification of the bts.properties file should not be attempted without Cisco TAC support or supervision.

The task is scheduled to run every day at midnight (12:00 a.m.). The task reads all the LERG6INS updates that need to be executed with a date prior to the current date and provision them. For example, assume that the today's date is 12-10-2005, and we add three lerg6_ins entries. The first one has a start time of 10-02-2005, the second has a start time of 12-10-2005, and the third has a start time of 12-15-2005. The task scheduled for today, which starts at 12-10-2005 at 12:00 a.m., checks the database with the entries loaded before or equal to the date 12-10-2005. The task fetches the two entries with the start times of 10-02-2005 and 12-10-2005 but ignores the entry with the start time of 12-15-2005. Hence, LERG6 is provisioned for the two entries with the start time of 10-02-2005 and 12-10-2005 but not for the entry with a start time of 12-15-2005.

System Context for LERG Support

Figure 3-3 illustrates the BTS 10200 context provisioning for LERG support.

Figure 3-3 *BTS 10200 Context Diagram of LERG Provisioning*



Configuring

The BTS 10200 LERG Support in OAMP feature configuration utilizes the **lergToCLI** script command to configure the system to create BTS 10200 specific CLI commands from the input LERG6.DAT or LERG6INS.DAT file. This script is available in the /opt/ems/bin directory.

Operator Interfaces

There are four types of operator interfaces available for LERG provisioning:

- The operator can provision the LERG data on the Call Agent using the one-time configuring mechanism which utilizes the `lergToCLI` command script.
- The operator can schedule provisioning of the LERG data on the Call Agent on a specific date in future using the `lergToCLI` command script.
- The operator can use CLI commands to add, change, delete, show, or audit the entries in the LERG table.
- The operator can use `show` or `delete` CLI commands to display or delete the currently scheduled LERG data which has not yet been sent to the Call Agent.

CLI Commands



Note

The BTS 10200 LERG Support in OAMP feature is developed to be used by the **lergToCLI** script command. The CLI option is provided for the sake of troubleshooting and debugging activities. The direct CLI commands using the CLI interface need to be used with discretion as simultaneous operations using the script and CLI interface might lead to complications.

You can use the **show** command to display the LERG6 data.

Examples:

```
show lerg6
```

```
show lerg6;ndc=469;ec=255;thousand_block =1
```

Users can use the **add** command to add the LERG6 data.

Example:

```
add lerg6 ndc=201;ec=007;lata=224;lata_name=NORTH JERSEY NJ;
thousand_block=A;coc_type=ATC;ssc=N;dind=Y;term_digits_ixc2eo=NA;term_digits_ixc2at=7;port
able=N;aocn=7228;ocn=7229;locality=XXXXXXXXXX;state=NJ;rc=XXXXXXXXXX;line_range_from=0000;
line_range_to=9999;switch_clli_code=NBWKNJ17DS1;sha=00;test_line_num=9921;thousand_block_p
ooling=N; lata_loc=224
```

The required attributes for **add lerg6** are:

```
aocn
dind
thousand_block
thousand_block_pooling
ec
portable
term_digits_ixc2eo
term_digits_ixc2at
ocn
ndc
switch_clli_code
```

You can use the **delete** command to delete the LERG6 data.

Example:

```
delete lerg6;ndc=469;ec=255
```

The required attributes for **delete** lerg6 are:

```
ec
ndc
thousadn_block
```

You can use the **show** command to display the currently scheduled LERG6INS data.

Examples:

```
show lerg6-ins

show lerg6-ins;start-time=2005-08-08 00:00:00

show lerg6-ins;id=20060808010100
```

You can use the **delete** command to delete the currently scheduled LERG6INS data.

Example:

```
delete lerg6-ins;id=2006010101000
```

The mandatory attributes for **delete** is “id”.

You can schedule the LERG6INS data update from the CLI using the **add** command.

Example:

```
add lerg6_ins noun=lerg6;verb=add; ndc=201;ec=252; start_time=2006-07-07 00 :00
:00;lata=224;lata_name=NORTH JERSEY NJ; thousand_block=3; coc_type=EOC; ssc=N; dind=Y;
term_digits_ixc2eo=NA;
term_digits_ixc2at=NA;portable=Y;aocn=4051;ocn=0558;locality=CRAGMERE;state=NJ;rc=CRAGMERE
;line_range_from=3000;line_range_to=3999;switch_clli_code=HCKNNJGRDS0;sha=01;test_line_num
=6299;thousand_block_pooling=Y; lata_loc=224
```

The mandatory attributes for **add** are “ec” and “start time”.

To clear all the LERG data, the following **clear** command can be used.

Examples:

```
clear lerg6

clear lerg6-ins
```

LERG6 Table

The Local Exchange Routing Guide (LERG) supports the current local exchange network within the NANP and shows planned changes in the network.

Table Name: LERG6

Table Containment Area: EMS, CA

Command Types

add, audit, change, clear, delete, help, show, sync



Caution

The **sync** command is a restricted command and is intended for repairing data only. Improper use may corrupt the database and disrupt call processing. Use with caution.

Usage Guidelines

Primary Key Token(s): DIGIT_STRING

Unique Key (Token(s): NDC, EC, THOUSAND_BLOCK

Syntax Description

AOCN	<p>Description: Mandatory for the add command. The Administrative Operating Company Number that identifies the company responsible for maintaining a particular record in the database underlying the Telcordia LERG Routing Guide.</p> <p>VARCHAR(4): 1-4 ASCII characters.</p> <p>Valid for Command: add, change</p> <p>Mandatory: add</p> <p>Possible Value: [1_4]</p> <p>Parser: AlphaNumericParser</p>
AUTO_REFRESH	<p>Description: Specifies whether to display cached data on the screen. Valid only for the show command.</p> <p>CHAR(1): Y/N (Default = Y).</p> <p>Y—Queries the database for the most current data.</p> <p>N—Queries the database for the most current data only if the cached data is unavailable.</p> <p>Valid for Command: show</p> <p>Default Value: Y</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>

COC_TYPE	<p>Description: Three-digit Central Office Code (COC).</p> <p>VARCHAR(3): 1-3 ASCII characters.</p> <p>Valid for Command: add, change</p> <p>Possible Value: [1_3]</p> <p>Parser: TextParser</p>
COUNTY	<p>Description: Two character abbreviation of the county where the associated record resides. This is completed only when more than one record using the same identifying keys exists within a state.</p> <p>VARCHAR(2): 1-2 ASCII characters.</p> <p>Valid for Command: add, change</p> <p>Possible Value: [1_2]</p> <p>Parser: DigitParser</p>
DIND	<p>Description: Specifies whether a code is dialable by the customer or operator.</p> <p>CHAR(1): Y/N.</p> <p>Y—Code is dialable.</p> <p>N—Code is not dialable.</p> <p>Valid for Command: add, change</p> <p>Mandatory: add</p> <p>Possible Value: [1_1]</p> <p>Parser: TextParser</p>
DISPLAY	<p>Description: Specifies what token information to display on the screen.</p> <p>VARCHAR(1024): 1-1024 (Default = all tokens are displayed). Permitted values are any valid token that can be shown for this command. Multiple tokens can be entered by separating with a comma.</p> <p>Valid for Command: show</p> <p>Possible Value: [1_1024]</p> <p>Parser: TextParser</p>
EC	<p>Description: Exchange code.</p> <p>VARCHAR(6): 1-6 ASCII characters.</p> <p>Valid for Command: add, delete, change, show, audit</p> <p>Mandatory: add, delete, change</p> <p>Possible Value: [1_3]</p> <p>Parser: DigitParser</p>

LATA	<p>Description: The three-digit number representing the geographical Local Access and Transport Area (LATA), or LATA-like code, in which the record is located. For example, 120 for Maine. There are five spaces for entering the LATA code—the last two digits are for the LATA sub-zone: full 5-digit LATA numbers are used only in Florida and represent Equal Access Exchange Areas (EAEAs).</p> <p>INTEGER: 100-99999 (Default = 99999) (3-5 numeric digits).</p> <p>Valid for Command: add, change</p> <p>Possible Value: [100_99999]</p> <p>Parser: DecimalParser</p>
LATA_LOC	<p>Valid for Command: add, change</p> <p>Possible Value: [1_5]</p> <p>Parser: TextParser</p>
LATA_NAME	<p>Description: The name of the LATA, or LATA-like code. For example, Maine.</p> <p>VARCHAR(20): 1-20 ASCII characters.</p> <p>Valid for Command: add, change</p> <p>Possible Value: [1_20]</p> <p>Parser: TextParser</p>
LIMIT	<p>Description: Specifies the number of rows to display on the screen.</p> <p>INTEGER: 1-100000000 (Default = 100000000).</p> <p>Valid for Command: show</p> <p>Default Value: 100000000</p> <p>Possible Value: [1_100000000]</p> <p>Parser: DecimalParser</p>
LINE_RANGE_FROM	<p>Description: The first four digits represent the starting number in this block of numbers.</p> <p>INTEGER(4): 1-4 numeric characters.</p> <p>Valid for Command: add, change</p> <p>Possible Value: [1_4]</p> <p>Parser: DigitParser</p>
LINE_RANGE_TO	<p>Description: The last four digits represent the last number in this block of numbers.</p> <p>INTEGER(4): 1-4 numeric characters.</p> <p>Valid for Command: add, change</p> <p>Possible Value: [1_4]</p> <p>Parser: DigitParser</p>

LOCALITY	<p>Description: Name of the locality served by this NXX code.</p> <p>VARCHAR(10): 1-10 ASCII characters.</p> <p>Valid for Command: add, change</p> <p>Possible Value: [1_10]</p> <p>Parser: TextParser</p>
MASTER	<p>Valid for Command: sync</p> <p>Mandatory: sync</p> <p>Possible Value: [1_10]</p> <p>Parser: TextParser</p>
NDC	<p>Description: Unique key: ndc, ec, dn-group. Foreign key: ndc+ec+office-code-index to the Exchange Code table. National Destination Code.</p> <p>VARCHAR(6): 1-6 ASCII characters.</p> <p>Valid for Command: add, delete, change, show, audit</p> <p>Mandatory: add, delete, change</p> <p>Possible Value: [1_3]</p> <p>Parser: DigitParser</p>
OCN	<p>Description: Operating Company Number (OCN). Identifies an NPA-NXX code-holder, switching entity company, or non-facility-based service providers such as resellers, billing service providers, and so forth.</p> <p>VARCHAR(4): 1-4 ASCII characters.</p> <p>Valid for Command: add, change</p> <p>Mandatory: add</p> <p>Possible Value: [1_4]</p> <p>Parser: AlphaNumericParser</p>
ORDER	<p>Description: Specifies whether to display data on the screen in a sorted order.</p> <p>VARCHAR(51200): 1-51200 (Default = all rows are displayed). Permitted values are any valid token that can be shown for this command. Multiple tokens can be entered by separating with a comma.</p> <p>Valid for Command: show</p> <p>Possible Value: [1_1024]</p> <p>Parser: TextParser</p>

PORTABLE	<p>Description: Mandatory for the add command. Specifies whether to port at least one line number in the NPA NXX.</p> <p>CHAR(1): Y/N.</p> <p>Y—Port at least one line number in the NPA NXX due to thousands-block-number pooling and/or service provider LNP.</p> <p>N—Do not port any numbers in the NPA NXX.</p> <p>Valid for Command: add, change</p> <p>Mandatory: add</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>
RC	<p>Description: Rate Center (RC). An RC is the approximate midpoint within a uniquely defined a Rate Exchange Area (REA) (although an RC can designate a geographic area as well), from which mileage measurements are determined.</p> <p>VARCHAR(10): 1-10 ASCII characters.</p> <p>Valid for Command: add, change</p> <p>Possible Value: [1_10]</p> <p>Parser: TextParser</p>
RC_TYPE	<p>Description: Identifies Rate Centers requiring special identification. The following are examples of types identifying a particular Rate Center:</p> <p>CHAR(1): U, S, Z. U—Unrestricted. An RC providing a range of telecommunications services that is not restricted to a specific function.</p> <p>S—Suburban Zone: defines large exchange areas. Suburban zones apply to large metropolitan areas and can include just the area around a city (for example, Pittsburgh suburban zones) or the city and its surrounding area (for example, Wheeling suburban zones). The exchange area must be large enough to warrant a subdivision of two or more suburban zones. Suburban zones are assigned a vertical and horizontal coordinate for use in measurements between RCs, suburban zones or zoned cities, in the same manner as RC vertical and horizontal coordinates.</p> <p>Z—Zoned City: defines a large exchange area usually encompassing a city (for example, New York City). Each zoned city is assigned a vertical and horizontal coordinate (identified as the Major Zone). In addition, the zoned city is sub-divided into two or more city zones. Vertical and horizontal coordinates are assigned to each city zone to use in the same manner as suburban zone vertical and horizontal coordinates.</p> <p>Valid for Command: add, change</p> <p>Possible Value: S, Z</p> <p>Parser: TextParser</p>

SHA	<p>Description: Switch Homing Arrangement (SHA) Indicator identifies the “homing” arrangement to be used for the NPA/NXX, relative to the entered 11-character switch. If populated with a value between 01 and 99 the combination of the 11 character switch identified and the 2 digit SHA indicator represent an alternative homing arrangement that must be previously established as an SH2 record within the routing database.</p> <p>INTEGER(2): 1-2 numeric characters.</p> <p>Valid for Command: add, change</p> <p>Possible Value: [1_2]</p> <p>Parser: DigitParser</p>
SSC	<p>Description: Special Service Code used in addition to the COC token identify special services provided by a Destination Code record.</p> <p>VARCHAR(5): 1-5 ASCII characters.</p> <p>Valid for Command: add, change</p> <p>Possible Value: [1_5]</p> <p>Parser: TextParser</p>
START_ROW	<p>Description: Specifies to begin displaying data on the screen at a specific row.</p> <p>INTEGER: 1-100000000 (Default = 1).</p> <p>Valid for Command: show</p> <p>Default Value: 1</p> <p>Possible Value: [1_100000000]</p> <p>Parser: DecimalParser</p>
STATE	<p>Description: This is the two-letter abbreviation that identifies a state, territory or province.</p> <p>VARCHAR(2): 1-2 ASCII characters.</p> <p>Valid for Command: add, change</p> <p>Possible Value: [1_2]</p> <p>Parser: TextParser</p>
SWITCH_CLLI_CODE	<p>Description: An 11-character descriptor of the switch provided by the owning Local Exchange Company for the purpose of routing calls. In most cases, this is the Common Language Location Identifier (CLLI) of the switch. For NXXs in the Caribbean and Bermuda NPAs, the SW IDENT this token is populated with SWCHxxUNKNO, where xx identifies one of the countries or territories in the Caribbean or Bermuda.</p> <p>VARCHAR(11): 1-11 ASCII characters.</p> <p>Valid for Command: add, change</p> <p>Mandatory: add</p> <p>Possible Value: [11_11]</p> <p>Parser: TextNotNullParser</p>

TARGET	<p>Description: Specifies the network element to receive the request.</p> <p>VARCHAR(5): 1-5 ASCII characters. Permitted values are:</p> <p>CA—Network identifier of a Call Agent.</p> <p>FSPTC (POTS/Tandem/Centrex Feature Server)—Network identifier of a specific Feature Server.</p> <p>FSAIN (AIN Feature Server)—Network identifier of AIN Feature Servers.</p> <p>Valid for Command: sync</p> <p>Mandatory: sync</p> <p>Possible Value: [1_10]</p> <p>Parser: TextParser</p>
TERM_DIGITS_ IXC2AT	<p>Description: Specifies the number of terminating digits required to route a call using a primary Access Tandem (AT) associated per LERG7SHA, LERG9 with a specified end office. This token is populated with either a 7 or 10. Seven terminating digits (NXX + line number) are needed if the terminating NPA-NXX is the same as the home NPA of the AT, where the AT serves a single NPA.</p> <p>Ten terminating digits (NPA + NXX + line number) are needed if the terminating NPA is different from the home NPA of the AT, where the AT serves two or more NPAs. The three-digit terminating NPA resolves the ambiguity between an NXX that is assigned in two or more NPAs served by a single Access Tandem.</p> <p>VARCHAR(2): 1-2 ASCII characters. Permitted values are:</p> <p>NA, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10.</p> <p>Valid for Command: add, change</p> <p>Mandatory: add</p> <p>Possible Value: NA, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10</p> <p>Parser: TextNotNullParser</p>
TERM_DIGITS_ IXC2EO	<p>Description: Specifies the number of terminating digits required to route a call direct to the EO where the NPA NXX resides. Both the EO token and the AT token must be provisioned. Blank entries are not allowed. An “NA” in either the EO or AT field means that the LSP provides for no direct trunking to the switching entity or point of termination (POT) in the LATA.</p> <p>VARCHAR(2): 1-2 ASCII characters. Permitted values are:</p> <p>NA, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10.</p> <p>Valid for Command: add, change</p> <p>Mandatory: add</p> <p>Possible Value: NA, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10</p> <p>Parser: TextNotNullParser</p>

TEST_LINE_NUM	Description: Specifies a test line that identifies the line number component of an NPA NXX (NPA-NXX = test line number). INTEGER(4): 1-4 numeric characters. Valid for Command: add, change Possible Value: [1_4] Parser: DigitParser
---------------	---

THOUSAND_BLOCK	<p>Description: Mandatory for add, change and delete. Specifies NPA NXX assignments and block ids. In LERG6 files, contains an “A” for NPA NXX records that are assigned to the Code Holder. In addition to the “A” record, NPA NXXs can have one or more numeric block ids (0-9) identified, as per the Thousands Block Pooling Administrative Guidelines. In the case of numeric block ids, this correlates to the 1000 line numbers that begin with the block id “1000.” (e.g. BLOCK ID 3 correlates to a range of 3000-3999). VARCHAR(3): 1_1.</p> <p>1—1000-1999 block ids 2—2000-2999 block ids 3—3000-3999 block ids 4—4000-4999 block ids 5—5000-5999 block ids 6—6000-6999 block ids 7—7000-7999 block ids 8—8000-8999 block ids 9—9000-9999 block ids</p> <p>Valid for Command: add, delete, change, show</p> <p>Mandatory: add, delete, change</p> <p>Possible Value: [1_1]</p> <p>Parser: TextParser</p>
----------------	---

THOUSAND_BLOCK_POOLING	<p>Description: Thousand (1000) Block Pooling indicator. CHAR(1): Y, N, S, I.</p> <p>Y— The NPA NXX is part of a pool of NXXs, within a given NPA, that are assigned 1000 lines at a time by the Pool Administrator (currently only applicable to the United States) to potentially different companies.</p> <p>N—The NXX is not publicly pooled and there is no information below the NXX level.</p> <p>S—The NXX is not publicly pooled, but seven digit routing is applied (for purposes of Intra Service Provider (SP) Pooling or other reason) to show the fully assigned NXX as “split” at the thousands block level. This can apply to any NPA within the NANP.</p> <p>I—Same as “S” with the addition of BIRRDS system controls regarding the “split” of the NXX into thousands blocks. The Code Holder must request this from the Pool Administrator. This applies only to NPAs that are in the Pool Administrator’s inventory in FCC regulated areas (U.S. and U.S. Territories).</p> <p>Valid for Command: add, change</p> <p>Mandatory: add</p> <p>Possible Value: Y, N, S, I</p> <p>Parser: TextParser</p>
------------------------	--

LERG6-INS Table

Lerg6-ins commands update the Lerg6 table on a monthly basis with LERG information from Telcordia. Lerg6-ins (in service) commands use the same tokens as the Lerg6 table. Lerg6-ins reports display changes made since the previous month's LERG. Use the add command to schedule a LERG6INS data update.

Table Name: LERG6_INS

Table Containment Area: EMS, CA

Command Types

add, change, clear, delete, help, show

Syntax Description

AOCN	<p>Description: Mandatory for the add command. The Administrative Operating Company Number that identifies the company responsible for maintaining a particular record in the database underlying the Telcordia LERG Routing Guide.</p> <p>VARCHAR(4): 1-4 ASCII characters.</p> <p>Valid for Command: add, change</p> <p>Possible Value: [1_4]</p> <p>Parser: AlphaNumericParser</p>
COC_TYPE	<p>Description: Three-digit Central Office Code (COC).</p> <p>VARCHAR(3): 1-3 ASCII characters.</p> <p>Valid for Command: add, change</p> <p>Possible Value: [1_3]</p> <p>Parser: TextParser</p>
COUNTY	<p>Description: Two character abbreviation of the county where the associated record resides. This is completed only when more than one record using the same identifying keys exists within a state.</p> <p>VARCHAR(2): 1-2 ASCII characters.</p> <p>Valid for Command: add, change</p> <p>Possible Value: [1_2]</p> <p>Parser: DigitParser</p>
DIND	<p>Description: Specifies whether a code is dialable by the customer or operator.</p> <p>CHAR(1): Y/N.</p> <p>Y—Code is dialable.</p> <p>N—Code is not dialable.</p> <p>Valid for Command: add, change</p> <p>Possible Value: [1_1]</p> <p>Parser: TextParser</p>

EC	<p>Description: Mandatory for add and delete. Foreign key: Exchange Code and National Destination Code tables. Exchange Code. Also referred as a COC.</p> <p>VARCHAR(6): 1-6 numeric characters.</p> <p>Valid for Command: add, change, show</p> <p>Mandatory: add</p> <p>Possible Value: [1_3]</p> <p>Parser: DigitParser</p>
ID	<p>Description: Primary key. Unique id for this ISDN D-channel profile. Assigned by a service provider.</p> <p>VARCHAR(16): 1-16 ASCII characters.</p> <p>Valid for Command: show, change, delete</p> <p>Mandatory: change, delete</p> <p>Possible Value: [1_20]</p> <p>Parser: DigitParser</p>
LATA	<p>Description: The three-digit number representing the geographical Local Access and Transport Area (LATA), or LATA-like code, in which the record is located. For example, 120 for Maine. There are five spaces for entering the LATA code—the last two digits are for the LATA sub-zone: full 5-digit LATA numbers are used only in Florida and represent Equal Access Exchange Areas (EAEAs).</p> <p>INTEGER: 100-99999 (Default = 99999) (3-5 numeric digits).</p> <p>Valid for Command: add, change</p> <p>Possible Value: [000_99999]</p> <p>Parser: DecimalParser</p>
LATA_LOC	<p>Valid for Command: add, change</p> <p>Possible Value: [1_5]</p> <p>Parser: TextParser</p>
LATA_NAME	<p>Description: The name of the LATA, or LATA-like code. For example, Maine.</p> <p>VARCHAR(20): 1-20 ASCII characters.</p> <p>Valid for Command: add, change</p> <p>Possible Value: [1_20]</p> <p>Parser: TextParser</p>
LINE_RANGE_FROM	<p>Description: The first four digits represent the starting number in this block of numbers.</p> <p>INTEGER(4): 1-4 numeric characters.</p> <p>Valid for Command: add, change</p> <p>Possible Value: [1_4]</p> <p>Parser: DigitParser</p>

LINE_RANGE_TO	<p>Description: The last four digits represent the last number in this block of numbers.</p> <p>INTEGER(4): 1-4 numeric characters.</p> <p>Valid for Command: add, change</p> <p>Possible Value: [1_4]</p> <p>Parser: DigitParser</p>
LOCALITY	<p>Description: Name of the locality served by this NXX code.</p> <p>VARCHAR(10): 1-10 ASCII characters.</p> <p>Valid for Command: add, change</p> <p>Possible Value: [1_10]</p> <p>Parser: TextParser</p>
NDC	<p>Description: Foreign key: Exchange Code and National Destination Code (NDC) tables. The NDC, which consists of the number plan area code (NPA) (first 3 digits of the destination code) combined with the COC.</p> <p>VARCHAR(6): 1-6 ASCII characters.</p> <p>Valid for Command: add, change, show</p> <p>Mandatory: add</p> <p>Possible Value: [1_3]</p> <p>Parser: DigitParser</p>
NOUN	<p>Description: Mandatory for add command. CLI noun.</p> <p>VARCHAR(64): 1-64 ASCII characters; any valid command noun.</p> <p>Valid for Command: add, change, show</p> <p>Mandatory: add</p> <p>Default Value: LERG6</p> <p>Possible Value: LERG6</p> <p>Parser: TextNotNullParser</p>
OCN	<p>Description: Operating Company Number (OCN). Identifies an NPA-NXX code-holder, switching entity company, or non-facility-based service providers such as resellers, billing service providers, and so forth.</p> <p>VARCHAR(4): 1-4 ASCII characters.</p> <p>Valid for Command: add, change</p> <p>Possible Value: [1_4]</p> <p>Parser: AlphaNumericParser</p>

PORTABLE	<p>Description: Mandatory for the add command. Specifies whether to port at least one line number in the NPA NXX.</p> <p>CHAR(1): Y/N.</p> <p>Y—Port at least one line number in the NPA NXX due to thousands-block-number pooling and/or service provider LNP.</p> <p>N—Do not port any numbers in the NPA NXX.</p> <p>Valid for Command: add, change</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>
RC	<p>Description: Rate Center (RC). An RC is the approximate midpoint within a uniquely defined a Rate Exchange Area (REA) (although an RC can designate a geographic area as well), from which mileage measurements are determined.</p> <p>VARCHAR(10): 1-10 ASCII characters.</p> <p>Valid for Command: add, change</p> <p>Possible Value: [1_10]</p> <p>Parser: TextParser</p>
RC_TYPE	<p>Description: Identifies Rate Centers requiring special identification. The following are examples of types identifying a particular Rate Center:</p> <p>CHAR(1): U, S, Z.</p> <p>U—Unrestricted. An RC providing a range of telecommunications services that is not restricted to a specific function.</p> <p>S—Suburban Zone: defines large exchange areas. Suburban zones apply to large metropolitan areas and can include just the area around a city (for example, Pittsburgh suburban zones) or the city and its surrounding area (for example, Wheeling suburban zones). The exchange area must be large enough to warrant a subdivision of two or more suburban zones. Suburban zones are assigned a vertical and horizontal coordinate for use in measurements between RCs, suburban zones or zoned cities, in the same manner as RC vertical and horizontal coordinates.</p> <p>Z—Zoned City: defines a large exchange area usually encompassing a city (for example, New York City). Each zoned city is assigned a vertical and horizontal coordinate (identified as the Major Zone). In addition, the zoned city is sub-divided into two or more city zones. Vertical and horizontal coordinates are assigned to each city zone to use in the same manner as suburban zone vertical and horizontal coordinates.</p> <p>Valid for Command: add, change</p> <p>Possible Value: S, Z</p> <p>Parser: TextParser</p>

SHA	<p>Description: Switch Homing Arrangement (SHA) Indicator identifies the “homing” arrangement to be used for the NPA/NXX, relative to the entered 11-character switch. If populated with a value between 01 and 99 the combination of the 11 character switch identified and the 2 digit SHA indicator represent an alternative homing arrangement that must be previously established as an SH2 record within the routing database.</p> <p>INTEGER(2): 1-2 numeric characters.</p> <p>Valid for Command: add, change</p> <p>Possible Value: [1_2]</p> <p>Parser: DigitParser</p>
SSC	<p>Description: Special Service Code used in addition to the COC token identify special services provided by a Destination Code record.</p> <p>VARCHAR(5): 1-5 ASCII characters.</p> <p>Valid for Command: add, change</p> <p>Possible Value: [1_5]</p> <p>Parser: TextParser</p>
START_TIME	<p>Description: Starting time for measurements summary. Enter all 19 ASCII characters as shown. Start-time must occur before end-time. Measurement items are available for the current and previous calendar days only (up to a maximum of 48 hours of events).</p> <p>DATE and TIME: YYYY-MM-DD HH:MM:SS.</p> <p>Valid for Command: add, change, show</p> <p>Mandatory: add</p> <p>Possible Value: [14_19]</p> <p>Parser: DateTimeParser</p>
STATE	<p>Description: This is the two-letter abbreviation that identifies a state, territory or province.</p> <p>VARCHAR(2): 1-2 ASCII characters.</p> <p>Valid for Command: add, change</p> <p>Possible Value: [1_2]</p> <p>Parser: TextParser</p>
SWITCH_CLLI_CODE	<p>Description: An 11-character descriptor of the switch provided by the owning Local Exchange Company for the purpose of routing calls. In most cases, this is the Common Language Location Identifier (CLLI) of the switch. For NXXs in the Caribbean and Bermuda NPAs, the SW IDENT this token is populated with SWCHxxUNKNO, where xx identifies one of the countries or territories in the Caribbean or Bermuda.</p> <p>VARCHAR(11): 1-11 ASCII characters.</p> <p>Valid for Command: add, change</p> <p>Possible Value: [11_11]</p> <p>Parser: TextNotNullParser</p>

TERM_DIGITS_ IXC2AT	<p>Description: Specifies the number of terminating digits required to route a call using a primary Access Tandem (AT) associated per LERG7SHA, LERG9 with a specified end office. This token is populated with either a 7 or 10. Seven terminating digits (NXX + line number) are needed if the terminating NPA-NXX is the same as the home NPA of the AT, where the AT serves a single NPA.</p> <p>Ten terminating digits (NPA + NXX + line number) are needed if the terminating NPA is different from the home NPA of the AT, where the AT serves two or more NPAs. The three-digit terminating NPA resolves the ambiguity between an NXX that is assigned in two or more NPAs served by a single Access Tandem.</p> <p>VARCHAR(2): 1-2 ASCII characters. Permitted values are: NA, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10.</p> <p>Valid for Command: add, change</p> <p>Possible Value: NA, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10</p> <p>Parser: TextNotNullParser</p>
TERM_DIGITS_ IXC2EO	<p>Description: Specifies the number of terminating digits required to route a call direct to the EO where the NPA NXX resides. Both the EO token and the AT token must be provisioned. Blank entries are not allowed. An “NA” in either the EO or AT field means that the LSP provides for no direct trunking to the switching entity or point of termination (POT) in the LATA.</p> <p>VARCHAR(2): 1-2 ASCII characters. Permitted values are:</p> <p>NA, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10.</p> <p>Valid for Command: add, change</p> <p>Possible Value: NA, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10</p> <p>Parser: TextNotNullParser</p>
TEST_LINE_NUM	<p>Description: Specifies a test line that identifies the line number component of an NPA NXX (NPA-NXX = test line number).</p> <p>INTEGER(4): 1-4 numeric characters.</p> <p>Valid for Command: add, change</p> <p>Possible Value: [1_4]</p> <p>Parser: DigitParser</p>

THOUSAND_BLOCK	<p>Description: Mandatory for add, change and delete. Specifies NPA NXX assignments and block ids. In LERG6 files, contains an “A” for NPA NXX records that are assigned to the Code Holder. In addition to the “A” record, NPA NXXs can have one or more numeric block ids (0-9) identified, as per the Thousands Block Pooling Administrative Guidelines. In the case of numeric block ids, this correlates to the 1000 line numbers that begin with the block id “1000.” (e.g. BLOCK ID 3 correlates to a range of 3000-3999).</p> <p>VARCHAR(3): 1_1.</p> <p>1—1000-1999 block ids</p> <p>2—2000-2999 block ids</p> <p>3—3000-3999 block ids</p> <p>4—4000-4999 block ids</p> <p>5—5000-5999 block ids</p> <p>6—6000-6999 block ids</p> <p>7—7000-7999 block ids</p> <p>8—8000-8999 block ids</p> <p>9—9000-9999 block ids</p> <p>Valid for Command: add, change, show</p> <p>Mandatory: add</p> <p>Possible Value: [1_1]</p> <p>Parser: TextParser</p>
----------------	---

THOUSAND_ BLOCK_POOLING	<p>Description: Thousand (1000) Block Pooling indicator.</p> <p>CHAR(1): Y, N, S, I.</p> <p>Y— The NPA NXX is part of a pool of NXXs, within a given NPA, that are assigned 1000 lines at a time by the Pool Administrator (currently only applicable to the United States) to potentially different companies.</p> <p>N—The NXX is not publicly pooled and there is no information below the NXX level.</p> <p>S—The NXX is not publicly pooled, but seven digit routing is applied (for purposes of Intra Service Provider (SP) Pooling or other reason) to show the fully assigned NXX as “split” at the thousands block level. This can apply to any NPA within the NANP.</p> <p>I—Same as “S” with the addition of BIRRDs system controls regarding the “split” of the NXX into thousands blocks. The Code Holder must request this from the Pool Administrator. This applies only to NPAs that are in the Pool Administrator’s inventory in FCC regulated areas (U.S. and U.S. Territories).</p> <p>Valid for Command: add, change</p> <p>Possible Value: Y, N, S, I</p> <p>Parser: TextParser</p>
VERB	<p>Description: Mandatory for add command. Command-line interface (CLI) verb.</p> <p>VARCHAR(64): 1-64 ASCII characters; any valid command verb.</p> <p>Valid for Command: add, change, show</p> <p>Mandatory: add</p> <p>Possible Value: ADD, CHANGE, DELETE</p> <p>Parser: TextNotNullParser</p>

LERG, TNS, and Additional SIP Extensions for CMS-MGC Separation

This section provides an overview of the LERG and TNS functionality.

LERG

The LERG database, provided by Telecordia, contains comprehensive data of all switches and their serving NPANXX numbers. If a switch's provisioning does not yield a route to a destination using normal Called Party Number analysis through the DIAL-PLAN, it should use LERG to figure out the remote switch that serves the NPANXX and route to the remote switch.



Note

LERG data provisioning is applicable only to ANSI/North America market.

This feature is intended to provide routing based on the LERG data supplied by Telecordia. If a called party number analysis does not yield a routing solution, LERG data is looked up to find a remote switch id and the call is routed based on the remote switch id.

There are two options to route to a remote switch from the BTS 10200. The first option is to provision a route to every remote switch in the dial-plan. The second option is to always route to a default destination/route.

The first option entails provisioning an entry in the routing data (dial-plan, destination, route, etc) for each of the remote switches in the country. That will yield a lot of entries in the routing data table of the BTS 10200 and complicates provisioning. The first option does not require any changes to the BTS 10200. The existing routing schema can be used to provision routes to remote switches.

The second approach would be to simply route to a default destination/route and put the onus on the adjoining switches.

The LERG implementation in the BTS 10200 uses the second option, the default routing approach. Also, the LERG data is looked up for digit validation and for finding call_type, but the remote_switch id is ignored. Whether to look up LERG data or not for validating digits and to find call_type is configurable. Configuration of default route and configuration of LERG data lookup is included in the dial-plan-profile. In the dial-plan-profile the default route is configured via the default_dest field and the LERG lookup is configured via the lerg_supp field. Both of the field can be configured independently.

The independent configuration of LERG and default route configuration in dial-plan-profile enables one to provision the default route with or without LERG lookup. This allows the default route to be configured in international markets where LERG data is not relevant. Additionally, the default route can be used in any routing scenario.

In addition a new call_type policy is added to enable routing based on the call_type.

[Figure 3-4](#) and [Figure 3-5](#) depict high-level views of routing calls originated from the trunk group and subscriber, respectively [Figure 3-6](#), depicts change needed to implement default_routing dial-plan routing. Here default_routing refers to both LERG lookup and looking up the default_dest from dial-plan-profile.

Figure 3-4 Routing of an Incoming Trunk Call

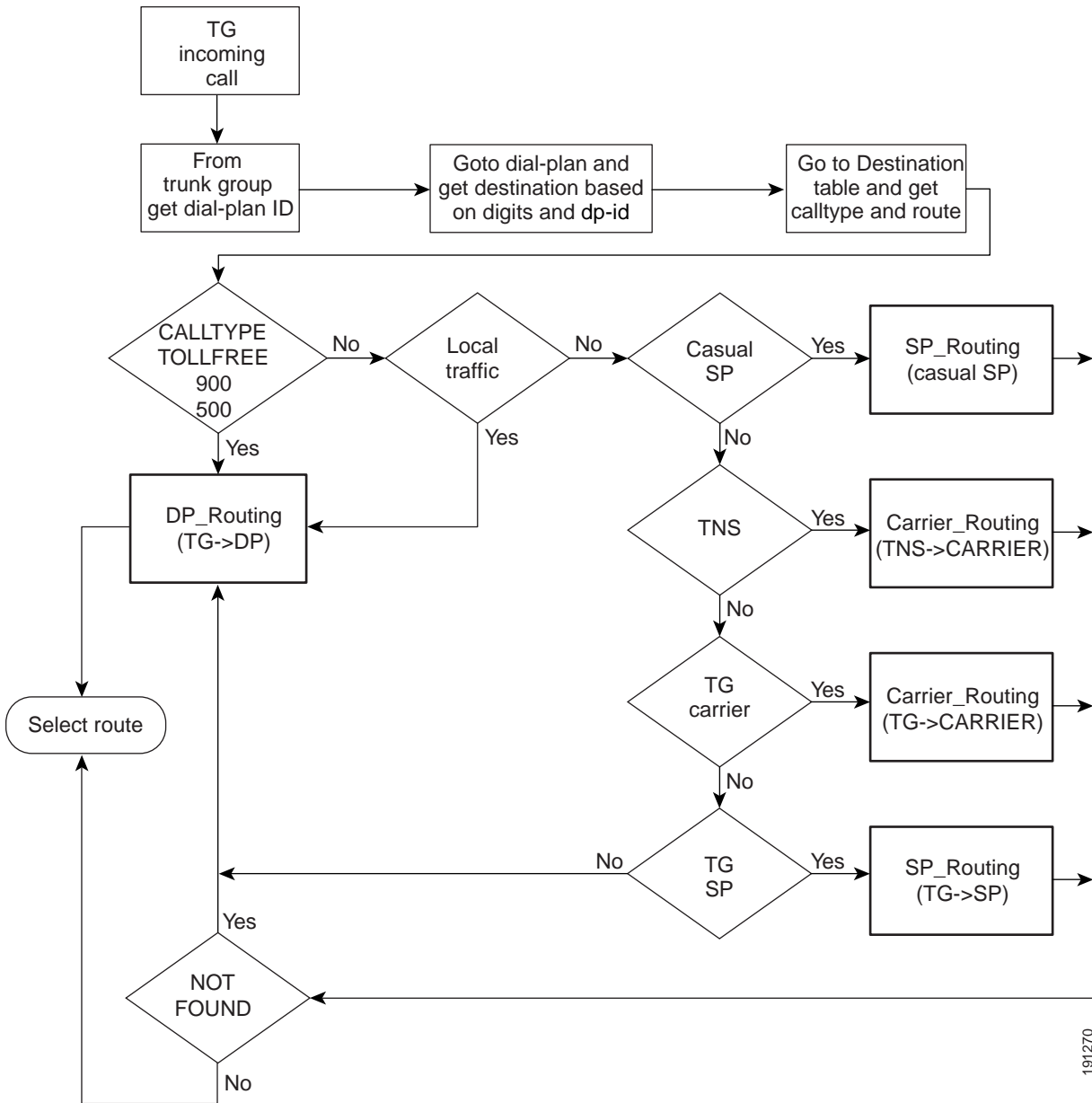


Figure 3-5 Routing on an Incoming Subscriber Call

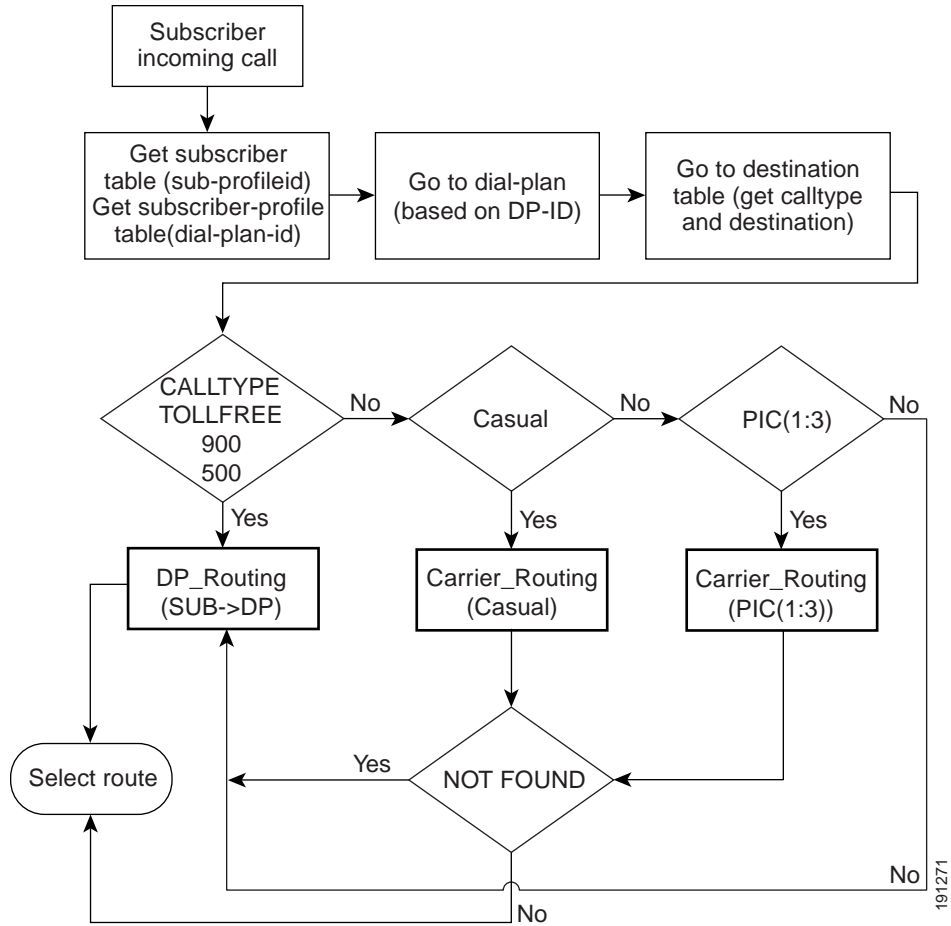
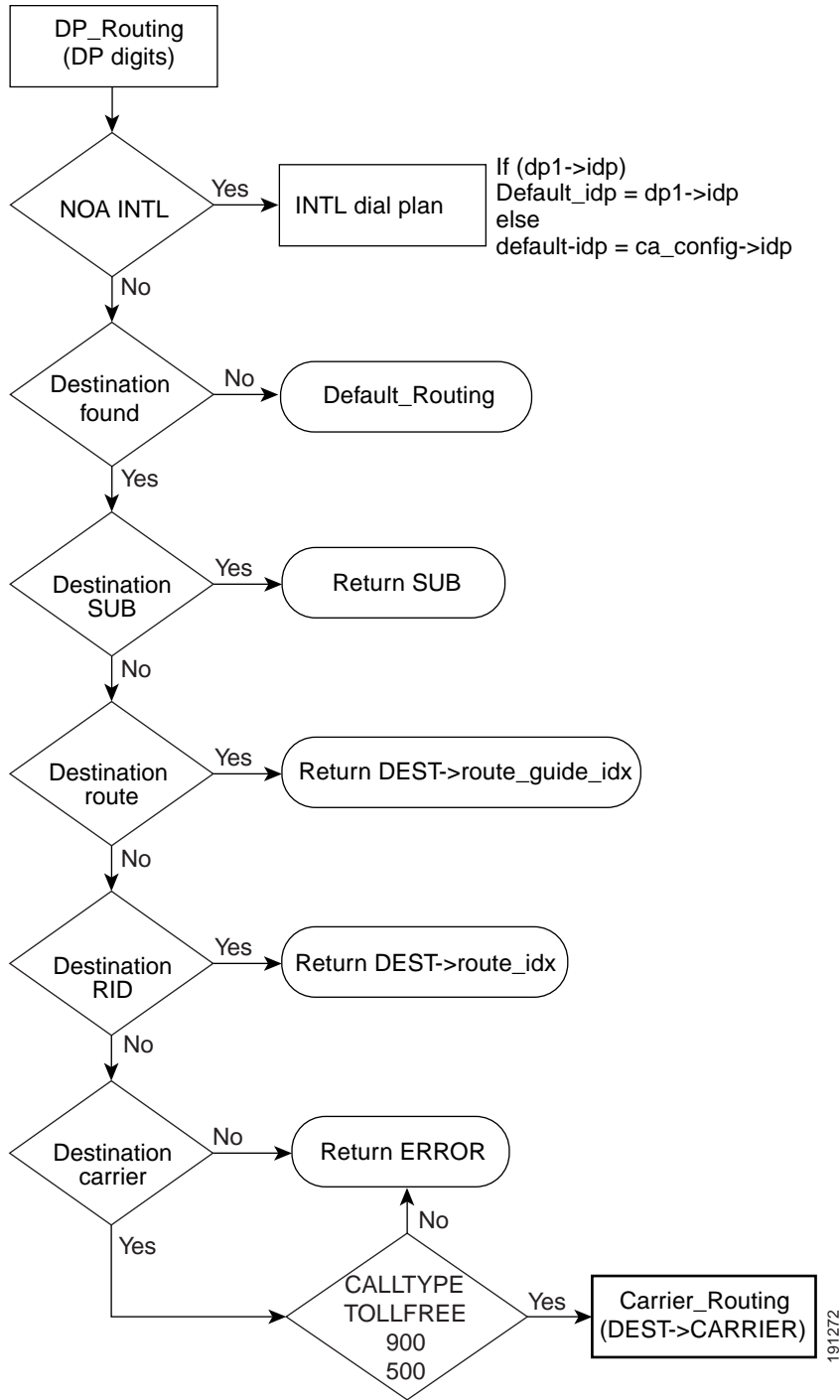


Figure 3-6 Default-Routing Dial-Plan Routing



TNS Routing Modifications

This section describes the TNS routing modifications.

Current TNS Routing Implementation

The current implementation of the BTS 10200 supports carrier based routing when it receives a TNS parameter in an IAM from PSTN network. In the current implementation, the carrier routing involves three steps: 1) find an entry in the dial-plan with Called Party digits, 2) find `call_type` based on the destination pointed to by the dial-plan entry, and 3) route using the `carrier_id`. Step 1 and 2 are mandatory. If an entry is not found in Step 1, the call is failed.

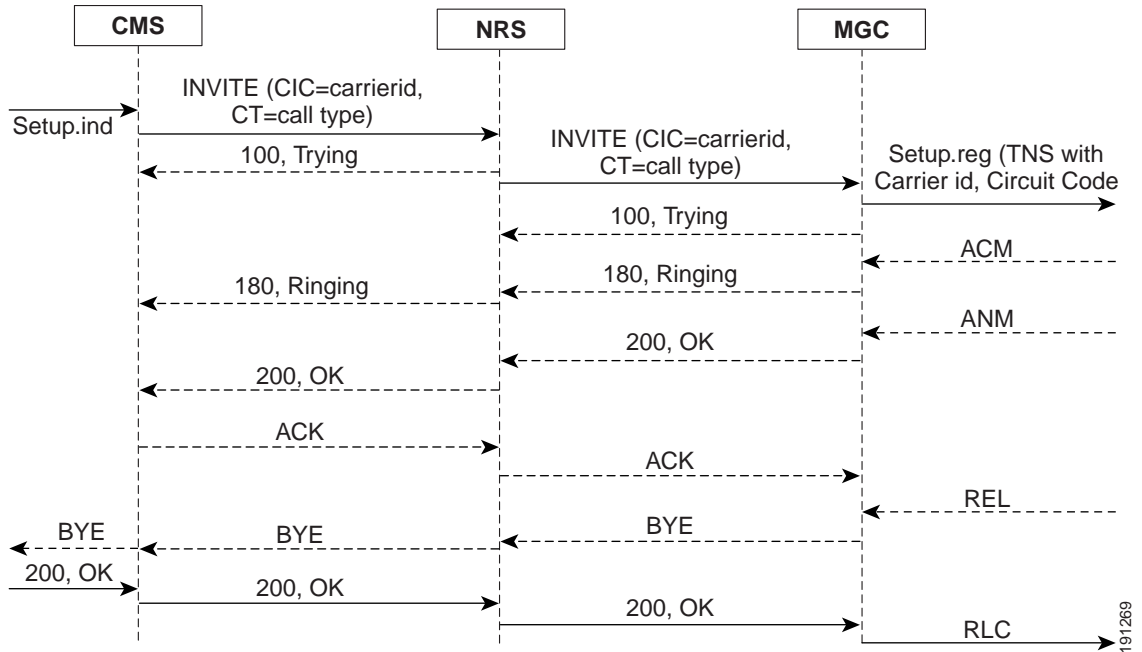
Modified TNS Routing Implementation

The modified TNS routing implementation, stemming from CMS and MGC separation, requires that the MGC routing analysis shall include `call_type` and `carrier_id` passed in INVITE from the NRS. With changes to TNS implementation derived from the MGC routing analysis requirement, the three basic carrier routing steps remain the same. But, with the addition of default routing, the provisioning of the dial-plan will be easier. Per the requirement, a `default_dest` field is added to the dial-plan-profile table so that if no dial-plan entry is found the `default_dest` in dial-plan-profile is used. With this, one needs to provide only one `default_dest`, no entries in the dial-plan are needed. As shown in the call_flow in [Figure 3-7](#), the CMS will pass the `call_type` and `carrier_id` from its route analysis; and the NRS, in turn, will pass it to the MGC.

One TG could be provisioned from the NRS to the MGC so that the NRS will channel all calls with `call_type` and `carrier_id` on that TG. Then on MGC, the dial-plan-profile is provisioned with the `default_dest` and no dial-plan entries are needed. The basic need of having a destination is still there, but the provisioning aspect is simplified with the introduction of default routing.

The change is applicable to all calls coming from either PSTN or NRS. However, `call_type` is expected only on the NRS to MGC trunk and Access Tandem trunk.

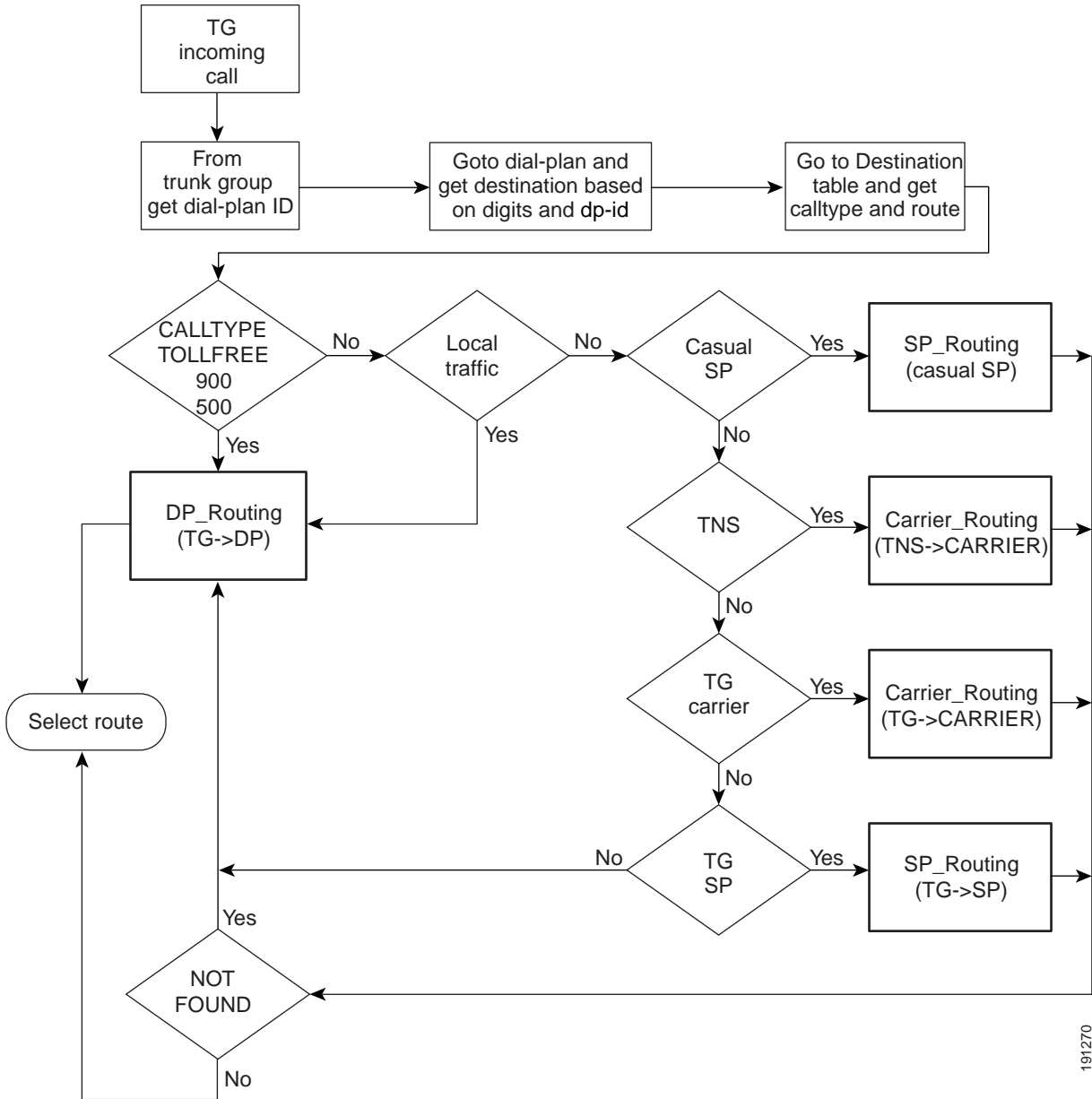
Figure 3-7 Call Flow Between a CMS and a MGC via a NRS



In the current design of the BCM, as shown in [Figure 3-8](#), every incoming call from the TG flows through the dial-plan and the TNS based routing is accomplished after a destination is found.

The TNS routing is modified to account for the call_type in the setup from NRS. The call_type in Call Data field is initialized with call_type from the Setup field. In addition, a new policy Circuit Code policy is added to route based on the Circuit Code.

Figure 3-8 TNS Routing and Modifications to the General Routing Flow



TNS and LERG Feature Use

The routing changes can be summarized as follows:

1. Validating and finding call type using LERG data
2. Route using default route
3. Call_type Policy
4. Circuit Code Policy
5. TNS routing modifications

The first item is only applicable to the North American Market.

The first three items in the list are relevant to both MGC and CMS deployments. The last two items on the list above are relevant only to a MGC deployment.

Each one of the above items can be configured independent of the others in the list. For example, default-routing can be provisioned without LERG data and call_type policy can be provisioned without TNS routing, etc. This independence enables to use default routing in any scenario. And also, use of Call Type and Circuit Code policies in any scenario.

Data Model Changes to Policy Routing Tables

A new default_policy concept and a new policy_profile table are introduced with the introduction of the call_type and circuit code policies.

The new policy_profile table has an 0 to M relationship with all policy tables. So, for every entry in a policy table there is an entry in the policy_profile table, but not vice-versa.

The policy_profile table serves two purposes:

1. It provides foreign key relationship to the route_guide table. In prior releases, policy associations with route guides was not explicitly enforced. With the introduction of this table, the route_guide entry is associated with the policy_profile entry. Every policy profile entry has an entry in the policy table associated with the policy_type. So, when policy is added to the route_guide, the database checks for its existence in policy_profile.
2. It also provides a place holder for the default policy and default policy types. This enables taking default course of action, if the policy table pointed to by the TYPE field in policy_profile doesn't have an entry. In the current design, only policy_call_type and policy_circuit_code will use this feature. For example, in the case of policy_call_type, for every policy, there can be multiple entries with different call_types. It is possible that we may not find an entry for the call_type of the call being routed. In that instance, the default_policy_type and default_policy_id fields from the policy_profile entry are used to route the call.



Note

Prior to adding policy_call_type entry there must be an entry in policy_profile with type=CTYPE.

Cluster Routing

A cluster is a group of CMS and MGC nodes that appear as a single logical CMS/MGC to the PSTN. For information on this feature, see the [“Cluster Routing” section on page 2-26](#).

Inter-Process Interface (SIA)

In support of CMS/MGC separation in the Packetcable network, several optional extensions are to be added to SIP INVITE request messages. These extensions carry between the CMS and an MGC, information that is significant to the PSTN. Prior to release 5.0 some of these types of extensions have been added to the BTS, including: cic, rn, and npdi. The new extensions for Release 5.0 include:

- “ct” – Call Type
- “noa” – Nature of Address
- “cpc” – Calling Party Category
- “dai” – Dial Around Indicator

It should be noted that the values that are being assigned to these new extensions are currently only for the ANSI/North American market. In the future the parameter values can be expanded to include values necessary for other world markets.

Also note that these extensions are generally added/encoded on the CMS and decoded on the MGC. This is because the MGC can utilize the extra information for routing and it interfaces to the PSTN, which may utilize the extra information provided in these headers. Because of this, the general rule of thumb when provisioning the associated softsw-tg-profile flags for CMS/MGC separation is for these parameters to be set to send_only on the CMS and rcv_only on the MGC.

Call Type, Nature of Address, and Dial Around Indicator are extensions to the user part of the Request URI of a SIP INVITE message. The Calling Party Category extension is added to the user part of the PAID header. These extensions will only be added to the initial SIP INVITE request.

Call Type

The possible values for call type are as follows:

```
"sac-500" / "sac-700" / "sac-900" / "svc-976" / "airlines" / "ambulance" / "ana" / "blv" / "business" /
"cut-thru" / "national-opr" / "da" / "da-toll" / "emergency" / "fire" / "info" / "inter-lata" / "international" /
"international-opr" / "international-wz1" / "lb-test" / "local" / "lrrn" / "mobile" / "national-opr" /
"non-emergency" / "opr" / "sip-trigger" / "police" / "premium" / "railways" / "relay" / "repair" /
"svc-code" / "tandem" / "test" / "tool" / "toll-free" / "traffic" / "wakeup" / "weather" / "unknown" /
"token"
```

In the decode direction the MGC BTS 10200 will receive incoming SIP INVITE messages with the call type parameter from an NRS. Note that the NRS does not know about this parameter and just blindly passes it through when it receives it from a CMS BTS 10200.

Nature Of Address

Nature of Address extension corresponds to the directory number in the Request URI. It is defined in PSTN call signaling protocols to classify the dialed number. It is used in conjunction with Call Type to classify the called party. Its possible values are:

```
“unknown” / “subscriber” / “national” / “inter-national” / “opr-req-sub” / “opr-req-nat” /
“opr-req-inter-nat” / “opr-req” / “test”
```

There is a special scenario for operator calls when the NOA=operator, the BCM will send the call information with an empty called party number. For the operator scenario which are routed via SIP, there should be a digman associated with the route to append "0" for NOA=operator.

Also, if you specify MATCH-STRING=^\$; then it will prefix "0" only if there are no digits.

Example:

```
change ROUTE ID=60999; TGN1_ID=60999; DEL_DIGITS1=0; DEL_DIGITS2=0; DEL_DIGITS3=0;
DEL_DIGITS4=0; DEL_DIGITS5=0; DEL_DIGITS6=0; DEL_DIGITS7=0; DEL_DIGITS8=0; DEL_DIGITS9=0;
DEL_DIGITS10=0; TG_SELECTION=RR; DNIS_DIGMAN_ID1=Zero_Operator; NEXT_ACTION=NONE; P1=0;
WT1=0; P2=0; WT2=0; P3=0; WT3=0; P4=0; WT4=0; P5=0; WT5=0; P6=0; WT6=0; P7=0; WT7=0; P8=0;
WT8=0; P9=0; WT9=0; P10=0; WT10=0; TYPE=ROUTE;
```

```
Add DIGMAN ID=Zero_Operator; RULE=1; MATCH_STRING=^; REPLACE_STRING=0&;
MATCH_NOA=OPERATOR; REPLACE_NOA=OPERATOR
```

Calling Party Category

The Calling Party Category (CPC) characterizes the type of user (or caller) who originates a call. It is used in the PSTN for call routing and call screening purposes. All of the possible CPU values are as follows:

“ordinary” / “prison” / “police” / “test” / “operator” / “payphone” / “unknown” / “hospital” / “cellular” / “cellular-roaming” / “non-coin-toll-free” / “coin-toll-free” / “genvalue”

The CPC extension values only refer to the common values defined in the ANSI/North America market. In ANSI ISUP, this information pertains to the OLI (Originating Line Information) parameter.

Dial Around Indicator

The dial-around-indicator (DAI) is added to the user part of the SIP Request URI of the first outgoing INVITE message. The dial-around indicator is added to indicate how the carrier ID code was derived. The DAI has the following values:

- “presub” - the CIC contains the caller's pre-subscribed carrier
- “presub-da” - the CIC contains the caller's dialed carrier-identification-code; the caller has a pre-subscribed carrier. The dialed value is the same as the subscriber's carrier.
- “presub-daUnkwn” - the CIC may contain either a caller dialed carrier-identification-code or the caller's pre-subscribed carrier
- “da” - the CIC contains the caller's dialed carrier-identification-code; the caller does not have a pre-subscribed carrier
- “CIC-chrgPty” - the CIC is the preferred carrier of the charged party
- “altCIC-chrgPty” - the CIC is the alternate carrier of the charged party
- “verbal-clgPty” - the CIC was delivered verbally by the calling party
- “verbal-chrgPty” - the CIC was delivered verbally by the charged party
- “emergency” - this is an emergency call
- “operator” - the carrier was selected by a network operator

EMS Changes

The Translate Trunk command is enhanced to add three additional parameters, `call_type`, `circuit_code` and `carrier_id` to translate calls coming with `carrier_id`, `circuit_code` and `call_type`. All three parameters are optional.

Examples:

```
translate trunk; tgn-id=2; call_type=local; called_dn=7034841301; circuit_code=1
translate trunk; tgn-id=2; carrier_id=288; call_type=local; circuit_code=1
translate trunk; tgn-id=2; carrier_id=288;
```

TNS Routing Modification

TNS routing is modified to enable the setting of `call_type` in the `call_data`. If `setup.ind` has a valid `call_type` the `calldata.call_type` is set to the value derived from `setup.ind`.

The Circuit Code Policy has been added to enable routing based on the circuit code. Additionally, the Call Type Policy has been added to enable routing based on the call type. Both policies can be configured and chained together.

Figure 3-9 and Figure 3-10 illustrate the two possible the carrier code policy can be configured. Figure 3-9 shows configuration of Circuit Code Policy that enables routing to a different route based on the Circuit Code. In Figure 3-10, the Call Type policy precedes the Circuit Code policy.

Figure 3-9 Circuit Code Policy

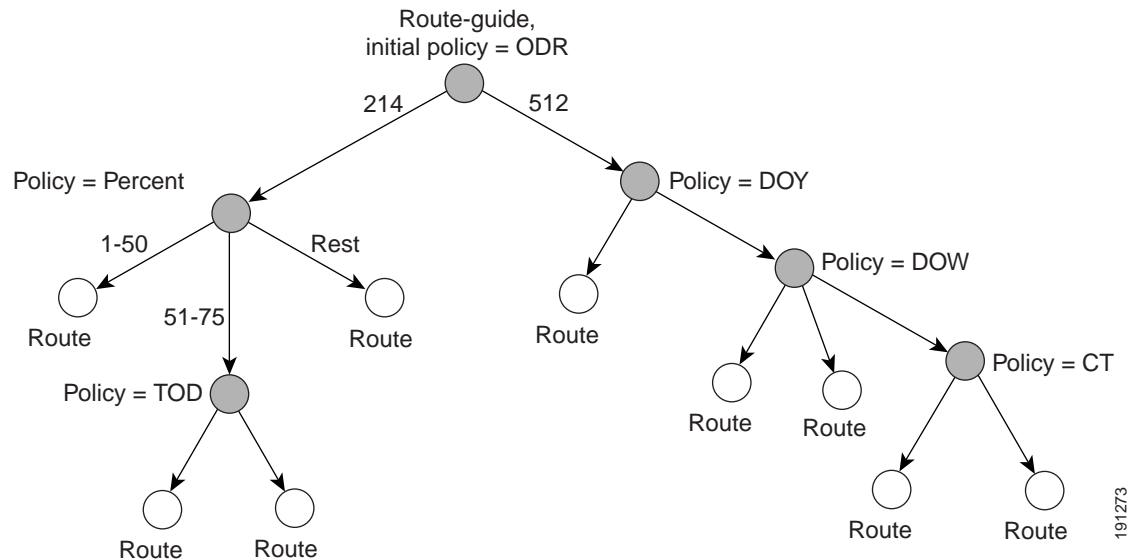
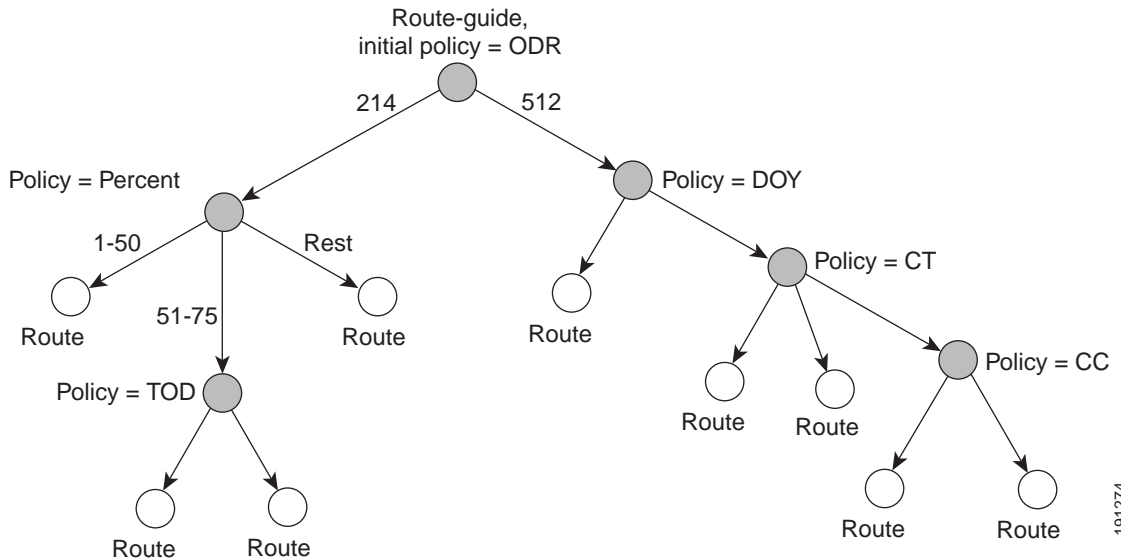


Figure 3-10 Call Type Policy and Circuit Code Policy Combination



Default Routing

Default routing is accomplished by using the new data field `DEFAULT-DEST-ID` in dial-plan-profile.

Use the following command to provide a default route.

Example:

```
change/add dia-plan-profile default_dest_id=dest1
```

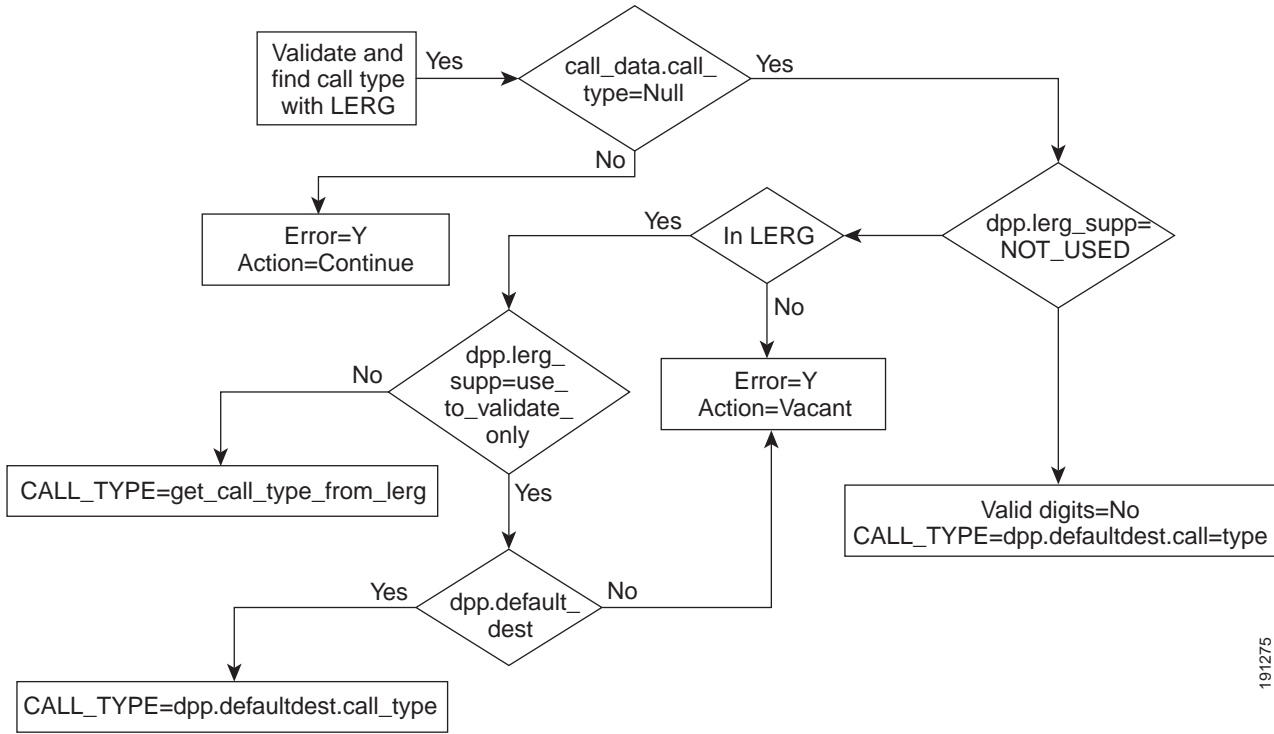
This command routes a call to default_dest “dest1”.

Validation Digits and Finding Call Type with LERG Data

The LERG data from Telecordia is used to validate the called party digits and to find call type based on the new dial-plan profile field `LERG_SUPP` values. If the value is `VALIDATE_AND_CALLTYPE`, the LERG data is used to validate the Called Party digits and call type evaluated based on the LERG entry. If the value is `USE-TO-VALIDATE-ONLY`, the LERG data is used to validate Called Party digits and call type is derived from the default-destination. If the value is `NONE`, the LERG table is not looked at. The `NANP-STATE` table is used to determine if the call_type is `INTL_WZ1` based on the state of the called party.

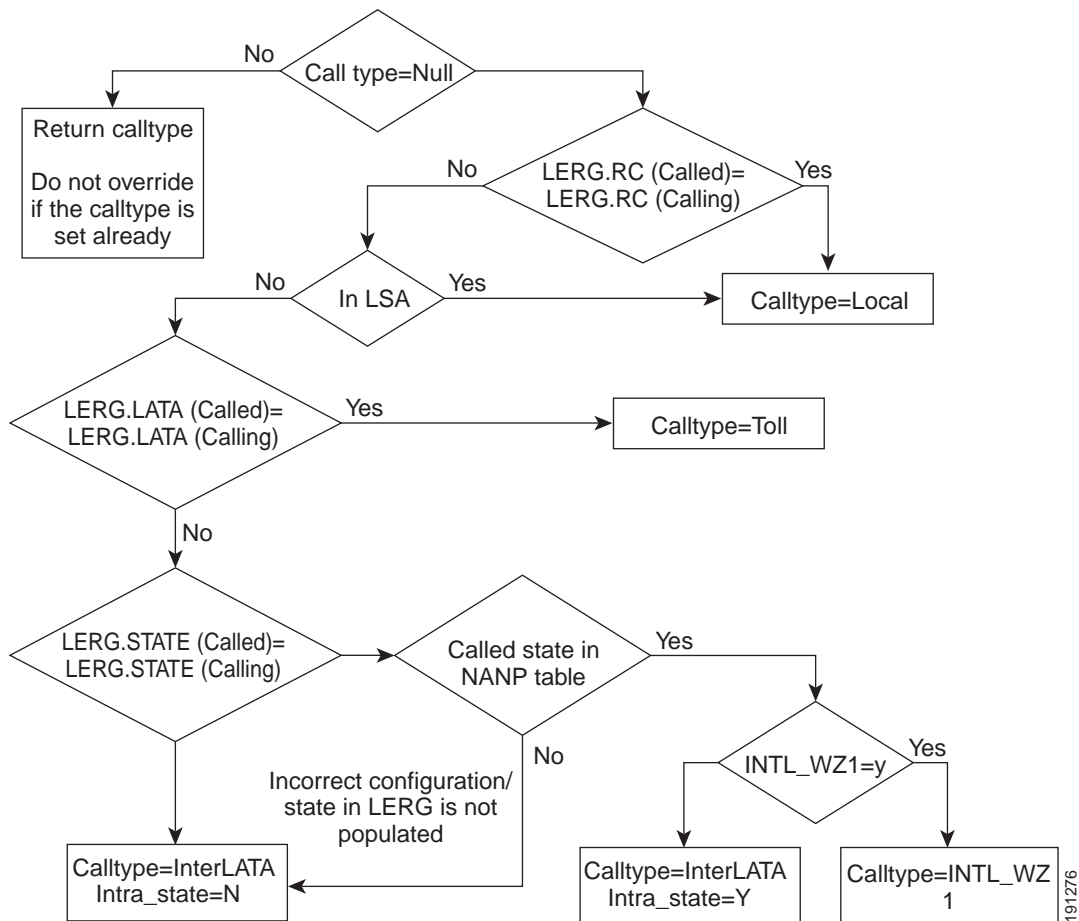
[Figure 3-11](#) shows the validation of digits using LERG. [Figure 3-12](#) shows the analysis of the call type using LERG.

Figure 3-11 Validating Digits Using LERG Data



191275

Figure 3-12 Call Type Analysis with LERG Data



Circuit Code Policy Routing

Circuit code policy routing enables route selection based on the Circuit Code from the TNS parameter.

Call Type Policy Routing

Call type policy routing enable the BTS 10200 to route calls based on the call type in scenarios like LNP and TOLL_FREE calls. In the case of TOLL_FREE or LNP query scenarios, the call type obtained before the query must be preserved by the BCM. The preserved pre-query call type and the new CALLTYPE policy is used to route the call. This enables the service provider to route the translated call to different trunk group based on the pre-query call type. Figure 3-13 and Figure 3-14 illustrate the processing and routing of LNP and TOLL_FREE calls, respectively.

Figure 3-13 Routing of an LNP Call

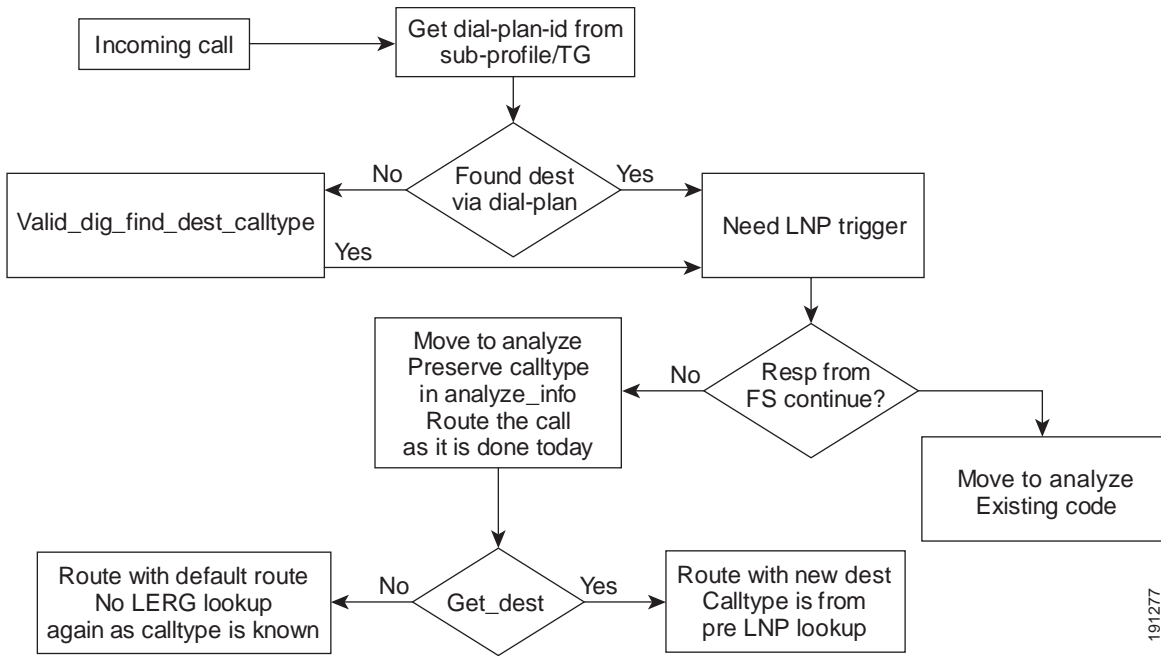
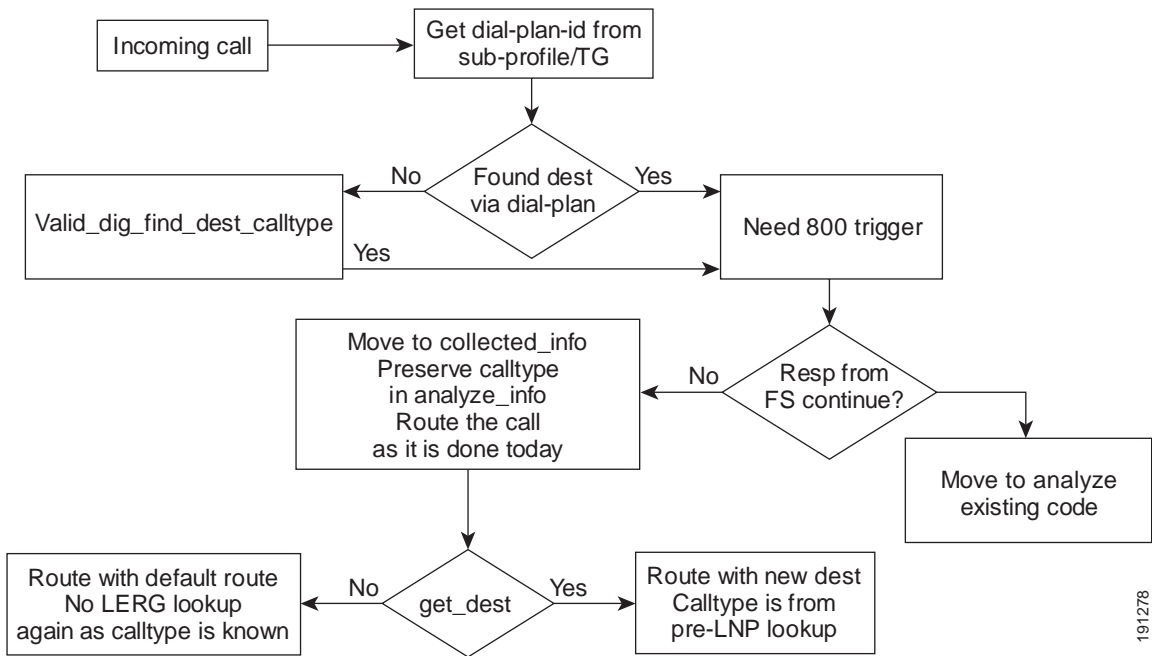


Figure 3-14 Routing of a TOLL_FREE Call



TNS Routing Modifications

No new provisioning is needed. To optimize routing, configure the following to circumvent looking up the dial-plan. For example, when a local call is routed from one CMS to another CMS, then the TG can be configured with a default_dest_id and default_dest_id is configured with Route_TYPE=SUB.

Example:

```
change/add dial-plan-profile id=dp69; default_dest_id=dest1
```

Setting Up LERG Validation and Call_Type

Use the following command to enable LERG data based called party digit validation and call_type.

Example:

```
change/add dial-plan-profile
LERG6_SUPP=NOT_USED/USED_TO_VALIDATE_ONLY/VALIDATE_AND_CALLTYPE
```

Setting Up Call_Type Policy

Use the following examples to configure the call_type policy.

Example:

To add a policy_profile:

```
add policy-profile id=carrier0288; type=ct; default-policy-type=route;
default-policy-id=rt1
```

Example:

To provision a policy_call_type:

```
add policy-call-type id=carrier0288; type=ct; call_type=local; policy_type=cc;
policy_id=CircuteCodePolicy1
```

```
add policy-call-type id=carrier0288; type=ct; call_type=INTL; policy_type=ROUTE;
policy_id=rt3
```



Note

Prior to adding policy_call_type entry there must be an entry in policy_profile with type=CTYPE.

Example:

To associate a policy_profile with a route_guide:

```
add route-guide id=rtguide1; policy_type=ct; policy_id=carrier0288
```


Setting Up Circuit_Code Policy

Use the following examples to configure circuit_code policy.

Example:

To add a policy_profile:

```
add policy-profile id=carrier0288; type=cc; default-policy-type=route;
default-policy-id=rt1
```

Example:

To provision a policy_call_type:

```
add policy-call-type id=carrier0288; type=cc; call_type=LOCAL; policy_type=cc;
policy_id=CircuteCodePolicy1
```

```
add policy_call_type id=carrier0288; type=CC; call_type=INTL; policy_type=ROUTE;
policy_id=rt3
```



Note

Prior to adding policy_call_type entry there must be an entry in policy_profile with type=CTYPE.

Example:

To associate a policy_profile with a route_guide:

```
Add route_guide id=rtguide1; policy_type=cc; policy_id=Carrier0288
```

Softswitch Trunk Group Profile Table

The Softswitch Trunk Group Profile (softsw-tg-profile) table holds all the information specific to a Softswitch trunk, such as id, protocol, indicators and echo suppression. The softsw-tg-profile record can be shared by multiple softswitch trunk groups. An ID must be created in this table before entries can be added to the Softswitch Trunk Group table.

Table Name: SOFTSW_TG_PROFILE

Table Containment: EMS, CA

Command Types

add, audit, change, delete, help, show, sync



Caution

The **sync** command is a restricted command and is intended for repairing data only. Improper use may corrupt the database and disrupt call processing. Use with caution.

Examples

```
add softsw-tg-profile id=softprf1; protocol-type=sip-t;
change softsw-tg-profile id=softprf1; send-cpn=n;
delete softsw-tg-profile id=softprf1;
show softsw-tg-profile id=softprf1;
```

Usage Guidelines

Primary Key: ID

Foreign Key: (SIP_TIMER_PROFILE_ID) references SIP_TIMER_PROFILE (ID)

Check Rule:

Add Rules: If protocol-type=sip-t; then sipt-isup-ver must be specified.

Add Rules: The sipt-isup-ver token must be defined in the SIPT ISUP Version Base table.

Delete Rules: ID cannot exist in any trunk-grp::tg-profile-id where tg-type=softsw.

Syntax Description

APPLY_USER_PRIVACY	<p>Description: Specifies whether to apply user privacy.</p> <p>CHAR(1): Y/N (Default = N).</p> <p>Y—If the originator requested privacy, aspects of the calling party information (such as the calling name and number in the From:header) in the initial outbound SIP INVITE is hidden. Privacy is requested when either the calling party name or number have presentation restrictions.</p> <p>N—User level privacy is not applied.</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: N</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>
AUTO_P_A_ID	<p>Description: Preliminary (proposed) Asserted Identity.</p> <p>CHAR(1): Y/N (Default = N).</p> <p>Y—Use PAID if received, else treat FROM header as PAID.</p> <p>N—Treat FROM header as PAID.</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: N</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>
AUTO_REFRESH	<p>Description: Specifies whether to display cached data on the screen.</p> <p>CHAR(1): Y/N (Default = Y).</p> <p>Y—Queries the database for the most current data.</p> <p>N—Queries the database for the most current data only if the cached data is unavailable.</p> <p>Valid for Command: show</p> <p>Default Value: Y</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>

DESCRIPTION	<p>Description: Described by the service provider.</p> <p>VARCHAR(64): 1-64 ASCII characters.</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Possible Value: [1_64]</p> <p>Parser: TextParser</p>
DISPLAY	<p>Description: Specifies what token information to display on the screen.</p> <p>VARCHAR(1024): 1-1024 (Default = all tokens are displayed). Permitted values are any valid token that can be shown for this command. Multiple tokens can be entered by separating with a comma.</p> <p>Valid for Command: show</p> <p>Possible Value: [1_1024]</p> <p>Parser: TextParser</p>
DIVERSION_ HEADER_SUPP	<p>Description: Specifies whether SIP Diversion Header is supported or not. This header conveys diversion information from other SIP user agents and proxies to the called user agent. This information can be used for enhanced features, including Unified Messaging, Third-Party voice mail, and Automatic Call Distribution (ACD). The most common use of the Diversion Header in the Cisco BTS 10200 Softswitch is for call forwarding features.</p> <p>CHAR(1): Y/N (Default = N).</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: Y</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>
DTMF_RELAY_ METHOD	<p>Description: Specifies which way to send an out-of-band DTMF Relay.</p> <p>VARCHAR(8): 1-8 ASCII characters. Permitted values are:</p> <p>NONE (Default)—Unsolicited DTMF Relay - Not supported.</p> <p>NOTIFY—DTMF Relay supported based on Subscribe/Notify Method.</p> <p>INFO—DTMF Relay supported based on INFO Method.</p> <p>Valid for Command: add, audit, change, show, sync</p> <p>Default Value: NONE</p> <p>Possible Value: NONE, NOTIFY, INFO</p> <p>Parser: TextNoCaseParser</p>

ENABLE_CPC_PARAM	<p>Description: Specifies whether to enable or disable processing of a calling party category (CPC) extension that is optionally added to the user part of a P-Asserted-ID header. The use-pai-hdr-for-ani token must be enabled for the enable-cpc-param token to be valid. If the use-pai-hdr-for-ani token is set to N, then the enable-cpc-param is ignored.</p> <p>VARCHAR(16): 1-16 ASCII characters. Permitted values are:</p> <p>SEND-ONLY—Send CPC in an outgoing INVITE message.</p> <p>RECV-ONLY—Process CPC from incoming INVITE messages.</p> <p>SEND-RECV—Send CPC in outgoing INVITE messages and process CPC on incoming INVITE messages,</p> <p>IGNORE (Default)—Disable CPC send and process.</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: IGNORE</p> <p>Possible Value: SEND_ONLY, RECV_ONLY, SEND_RECV, IGNORE</p> <p>Parser: TextNoCaseParser</p>
<hr/>	
ENABLE_CT_PARAM	<p>Description: Specifies whether to enable or disable the processing of a call type (CT) extension that is optionally added to the user part of the SIP Request URI.</p> <p>VARCHAR(16): 1-16 ASCII characters. Permitted values are:</p> <p>SEND-ONLY—Send CT in outgoing INVITE message.</p> <p>RECV-ONLY—Process CT from incoming INVITE messages.</p> <p>SEND-RECV—Send CT in outgoing INVITE messages and process CT on incoming INVITE messages.</p> <p>IGNORE (Default)—Disable CT send and process.</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: IGNORE</p> <p>Possible Value: SEND_ONLY, RECV_ONLY, SEND_RECV, IGNORE</p> <p>Parser: TextNoCaseParser</p>

ENABLE_DAI_PARAM	<p>Description: Specifies whether to enable or disable the processing of the dial around indicator (DAI) extension that is optionally added to the user part of the SIP Request URI. Please note however that in a Packetcable environment, CMSS 1.5 makes it mandatory to process DAI.</p> <p>VARCHAR(16): 1-16 ASCII characters. Permitted values are:</p> <p>SEND-ONLY—Send DAI in outgoing INVITE messages.</p> <p>RECV-ONLY—Process DAI from incoming INVITE messages.</p> <p>SEND-RECV—Send DAI in outgoing INVITE messages and process DAI on incoming INVITE messages.</p> <p>IGNORE (Default)—Disable DAI send and process.</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: IGNORE</p> <p>Possible Value: SEND_ONLY, RECV_ONLY, SEND_RECV, IGNORE</p> <p>Parser: TextNoCaseParser</p>
ENABLE_EM_EVENTS	<p>Description: Specifies whether to generate a Billing Correlation ID (BCID).</p> <p>CHAR(1): Y/N (Default = N).</p> <p>Y—Generate BCID if em-events-enabled=Y.</p> <p>N—Do not generate BCID. Assumes CMSS, events not generated.</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: N</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>
ENABLE_ES_EVENTS	<p>Description: Specifies whether to send or suppress additional Electronic Surveillance messages towards a DF server for calls that are traversing through a softswitch trunk group.</p> <p>CHAR(1): Y/N (Default = N).</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: N</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>

ENABLE_NOA_PARAM	<p>Description: Specifies whether to enable or disable processing of the Nature of Address (NOA) extension that is optionally added to the user part of a SIP Request URI.</p> <p>VARCHAR(16): 1-16 ASCII characters. Permitted values are:</p> <p>SEND-ONLY—Send NOA in outgoing INVITE messages.</p> <p>RCV-ONLY—Process NOA from incoming INVITE messages.</p> <p>SEND-RCV—Send NOA in outgoing INVITE messages and process NOA on incoming INVITE messages.</p> <p>IGNORE (Default)—Disable NOA send and process.</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: IGNORE</p> <p>Possible Value: SEND_ONLY, RCV_ONLY, SEND_RECV, IGNORE</p> <p>Parser: TextNoCaseParser</p>
ENABLE_P_DCS_BILLING_INFO_HDR	<p>Description: Specifies whether to enable PacketCable Distributed Call Signaling (DCS) billing in a SIP information header.</p> <p>CHAR(1): Y/N (Default = N).</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: N</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>
ENABLE_P_DCS_LAES_HEADER	<p>Description: Specifies whether to send surveillance information as defined in Section 8 of RFC 3603 if required, when it cannot be performed on the switch. Requires that the remote SIP entity interfacing with the SIP trunk support surveillance procedures.</p> <p>CHAR(1): Y/N (Default = N).</p> <p>N—Do not send surveillance information.</p> <p>Y—Send surveillance information.</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: N</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>

ENABLE_P_DCS_ OSPS_HEADER	<p>Description: Specifies whether to enable PacketCable DCS Operator Services Position System (OSPS) associated call support on a softswitch trunk group. If supported, calls include: busy line verification, emergency interrupt, and 911 operator ringback.</p> <p>When this flag is set and an OSPS related request is made, . If this flag is not set, BTS will not send outgoing SIP requests or accept incoming SIP requests that are OSPS related.</p> <p>CHAR(1): Y / N (Default = N).</p> <p>Y—Include a P-DCS OSPS header in the outgoing INVITE or UPDATE messages as defined in RFC-3603 for an OSPS related request.</p> <p>N—Do not include outgoing SIP requests or accept incoming SIP requests that are OSPS related.</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: N</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>
ENABLE_SIP_ TRIGGER	<p>Description: Specifies whether to enable SIP triggers on a SIP trunk group.</p> <p>CHAR(1): Y / N (Default = N).</p> <p>Y—Send outgoing calls to the Application server and incoming calls from the Application server are treated as SIP trigger calls.</p> <p>N—Outgoing calls are not sent to the Application server and incoming calls from the Application server are treated as regular calls (non-SIP-trigger calls).</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: N</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>
GTD_MODE	<p>Description: Specifies whether to use the compact (default) or verbose mode to encode messages for the SIP-T/GTD trunk group.</p> <p>VARCHAR(8): 1-8 ASCII characters. Permitted values are:</p> <p>COMPACT (Default)</p> <p>VERBOSE</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: COMPACT</p> <p>Possible Value: COMPACT, VERBOSE</p> <p>Parser: TextParser</p>

GTD_PARMS	<p>Description: Mandatory if protocol-type=sip-gtd. Specifies a comma-separated list of Generic Transparency Descriptor (GTD) parameters enabled for this profile. The parameters are parsed against a static table, called the GTD Parameter Values table, which lists all the valid GTD parameters, including the special case parameter ALL. In the DBM of the Call Agent, this comma-separated string is converted into a series of boolean flags, one for each GTD parameter. The Call Agent accesses each individual flag as it builds a GTD attachment.</p> <p>VARCHAR(500): 3-500 ASCII characters. For example:</p> <p>ALL—use all GTD parameters (or)</p> <p>CPN, CGN, CIC, CPC, BCI (comma-separated list)</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: ALL</p> <p>Possible Value: [1_500]</p> <p>Parser: TextNoCaseParser</p>
HOP_COUNTER_MAX	<p>Description: Applies only to received SIP Invite messages that are not SIP-T and contain a max-forwards value in which the max-forwards is scaled down to build the hop counter. If the hop counter derived from the max-forwards is greater than this value, it is set to this value. This value acts as a ceiling for the derived hop counter value.</p> <p>INTEGER: 10-20 (Default = 20).</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: 20</p> <p>Possible Value: [10_20]</p> <p>Parser: DecimalParser</p>
HOP_COUNTER_SUPP	<p>Description: Used for received SIP Invite messages that are not SIP-T and contain a max-forwards value. The default sets the hop counter based on the received max-forwards value. If this flag is set to N, the hop counter field is not populated using the max-forwards value.</p> <p>CHAR(1): Y/N (Default = Y).</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: Y</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>
ID	<p>Description: Primary key. Unique ID for this trunk group profile.</p> <p>VARCHAR(16): 1-16 ASCII characters.</p> <p>Valid for Command: add, change, show, delete, audit, sync</p> <p>Mandatory: add, change, delete</p> <p>Possible Value: [1_16]</p> <p>Parser: TextParser</p>

INBAND_TONE_AVAILABLE	<p>Description: Send release or provide tone/announcement.</p> <p>CHAR(1): Y/N (Default = Y).</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: Y</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>
LIMIT	<p>Description: Specifies the number of rows to display on the screen.</p> <p>INTEGER: 1-100000000 (Default = 100000000).</p> <p>Valid for Command: show</p> <p>Default Value: 100000000</p> <p>Possible Value: [1_100000000]</p> <p>Parser: DecimalParser</p>
MASTER	<p>Valid for Command: sync</p> <p>Mandatory: sync</p> <p>Possible Value: [1_10]</p> <p>Parser: TextParser</p>
MAX_FORWARDS	<p>Description: Specifies when an outbound SIP Invite message requires an initial maximum forwards value.</p> <p>INTEGER: 10-80 (Default = 70).</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: 70</p> <p>Possible Value: [4_80]</p> <p>Parser: DecimalParser</p>
ORDER	<p>Description: Specifies whether to display data on the screen in a sorted order.</p> <p>VARCHAR(1024): 1-1024 (Default = all rows are displayed). Permitted values are any valid token that can be shown for this command. Multiple tokens can be entered by separating with a comma.</p> <p>Valid for Command: show</p> <p>Possible Value: [1_1024]</p> <p>Parser: TextParser</p>
PLATFORM_STATE	<p>Description: Audits a shared memory database.</p> <p>VARCHAR(7): 1-7 ASCII characters. Permitted values are:</p> <p>ACTIVE (Default) - System is currently running.</p> <p>STANDBY.</p> <p>Valid for Command: sync, audit</p> <p>Default Value: ACTIVE</p> <p>Possible Value: ACTIVE, STANDBY</p> <p>Parser: TextParser</p>

PRACK_FLAG	<p>Description: Specifies if an Invite messages sent on this trunk group require reliable provisional responses. If yes, provisional responses like alerting are delivered. Used with SIP-T.</p> <p>CHAR(1): Y/N (Default = N).</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: N</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>
PROTOCOL_TYPE	<p>Description: Specifies the type of signaling for this trunk group. It controls the message type sent between two Cisco BTS 10200 Softswitches. For example, if the protocol-type is SIP-T, then the Cisco BTS 10200 Softswitch sends a SIP-T message, which is a normal SIP ASCII message plus an ISUP MIME attachment. In this case, the origination type can be ISDN, SS7, CAS, MGCP, and so forth. The origination type does not matter. However, if the protocol-type is SIP, then the Cisco BTS 10200 Softswitch sends only an ASCII SIP message without an ISUP MIME attachment.</p> <p>VARCHAR(9): 1-9 ASCII characters. Permitted values are:</p> <p>SIP—Signaling via the Session Initiation Protocol (SIP) multimedia sessions across the Internet.</p> <p>SIP-T—Signaling using SIP-T protocol. SIP-T is an inter Call Agent protocol; SIP-GTD protocol is a normalized inter Call Agent protocol.</p> <p>SIP-T—Signaling using both the SIP-T and SIP-GTD protocol types. SIP-T is an inter Call Agent protocol; SIP-GTD protocol is a normalized inter Call Agent protocol.</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Mandatory: add</p> <p>Possible Value: SIP, SIP_T</p> <p>Parser: TextParser</p>
RECV_3XX_USE_CF_METHOD	<p>Description: Specifies how a received 3xx response is handled. Applies only to 3xx responses received with a contact header containing a different number than the called party number in the request, where the domain name identifies the Cisco BTS 10200 Softswitch that sent the request. Feature provisioning is required before enabling this token.</p> <p>CHAR(1): Y/N (Default = Y).</p> <p>Y—Handle as a network-based reroute.</p> <p>N—Handle as a call forwarding request from the switch sending the 3xx.</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: Y</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>

REDIRECT_SUPPORTED	<p>Description: Specifies if the Cisco BTS 10200 Softswitch honors a 3xx class, such as a redirection response for an Invite message sent by the Cisco BTS 10200 Softswitch.</p> <p>VARCHAR(32): 1-32 ASCII characters. Permitted values are:</p> <p>VALID-DOMAINS-ONLY (Default)—If the host name field in the SIP URI of a 3XX contact used for call redirection does not represent this Cisco BTS 10200 Softswitch or a Cisco BTS 10200 Softswitch SIP trunk, then the call is redirected using the SIP trunk used on the previous call redirection. If there was not a previous call redirection, then the SIP trunk that sent the initial Invite is used. If the profile of the selected SIP trunk restricts redirection to only valid domains, then this redirection is blocked and the next contact is tried. Otherwise, it is redirected and the contact URI is used as the request URI of the redirected call.</p> <p>ALL-DOMAINS—Redirects to any allowed domain.</p> <p>NONE—No redirects allowed.</p> <p>Valid for Command: add, audit, change, show, sync</p> <p>Mandatory: add</p> <p>Default Value: VALID_DOMAINS_ONLY</p> <p>Possible Value: ALL_DOMAINS, NONE, VALID_DOMAINS_ONLY</p> <p>Parser: TextParser</p>
REFER_ALLOWED	<p>Description: Call Transfer allowed on an SS trunk.</p> <p>CHAR(1): Y / N (Default = N).</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: N</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>
REFERRED_BY_REQD_ON_REFERER	<p>Description: Specifies whether a “referred-by” header is required on REFER messages.</p> <p>CHAR(1): Y / N (Default = N).</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: N</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>

REPLACES_ALLOWED	<p>Description: Specifies whether to accept or reject received INVITE messages with a “replaces” header. Refer to RFC 3891 for more information.</p> <p>CHAR(1): Y / N (Default = N).</p> <p>Y—Accept received INVITE messages with a “replaces” header.</p> <p>N—Reject received INVITE messages with a “replaces” header.</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: N</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>
SCALE_FACTOR	<p>Description: Used for conversions between hop counter and max-forwards values; allows no-conversion, one-half, one-third, and one-quarter conversion factors. The default provides a scale relative to the maximum values: if the hop counter is 20, a scale factor of 4 converts to a max-forwards value of 80. Using the default means no conversion.</p> <p>INTEGER: 1-4 (Default = 1).</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: 1</p> <p>Possible Value: [1_4]</p> <p>Parser: DecimalParser</p>
SEND_302_ON_CF	<p>Description: Specifies whether “Send 302 on Call Forwarding” is supported.</p> <p>CHAR(1): Y/N (Default = N).</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: N</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>
SEND_3XX_DOMAIN_NAME	<p>Description: Specifies whether to apply the domain name in the contact header when sending a 3XX response. Applies only if send-302-on-cf is enabled and “call forwarding unavailable” is locally invoked and configured to send a 3XX SIP response. This token does not apply if the Cisco BTS 10200 Softswitch proxies a received 3XX response. For proxies, the domain name in the contact header of the 3XX received is preserved.</p> <p>VARCHAR(64): 1-64 ASCII characters.</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Possible Value: [1_64]</p> <p>Parser: DomainParser</p>

SEND_CIC_PARAM	<p>Description: Specifies whether the CIC parameter is included in the request URL for outbound SIP calls.</p> <p>CHAR(1): Y/N (Default = Y)</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: Y</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>
SEND_FULL_E164	<p>Description: When enabled, all SIP phone numbers contained in SIP messages sent from the Cisco BTS 10200 Softswitch that have an NOA of national significance are represented as fully qualified E.164 numbers prefixed with the local country code and plus sign. This conforms to IETF RFC 3398 Section 12.1. When disabled, national numbers are sent without a country code and plus sign prefix. Numbers of international significance are always sent with a plus sign and country code regardless of this token setting. The Home Country code is defined in the Call Agent Configuration table.</p> <p>CHAR(1): Y/N (Default = N).</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: N</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>
SEND_LAES_IN_RESPONSE	<p>Description: Specifies whether the Cisco BTS 10200 Softswitch can include a Lawfully Authorized Electronic Surveillance (LAES) in a 183 Alerting message if a PacketCable call content Internet Access Point (IAP) for Real-Time Transport Protocol (RTP) duplication is not found.</p> <p>CHAR(1): Y/N (Default = N).</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: N</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>
SEND_PHONE_CONTEXT_PARAM	<p>Description: Specifies whether to tag the local telephone number with a telephone context parameter. If send-full-e164 is set, this token is significant for number that cannot be represented in an E.164 format, such as 911.</p> <p>CHAR(1): Y/N (Default = N).</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: N</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>

SEND_SIP_181_RESP	<p>Description: Specifies whether the Cisco BTS 10200 Softswitch transmits a 181 response message to a UAC when the terminating side of the Cisco BTS 10200 Softswitch forwarded the call.</p> <p>CHAR(1): Y/N (Default = N)</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: N</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>
SEND_STD_TRK_GRP_URI	<p>Description: Specifies whether to use draft IETF IP Telephony (iptel) Trunk Group (draft-ietf-iptel-trunk-group) defined trunk group parameters when an INVITE request is received and the trunk-sub-grp-type=TGID.</p> <p>CHAR(1): Y/N (Default = N).</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: N</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>
SESSION_TIMER_ALLOWED	<p>Description: Specifies whether a session timer is allowed.</p> <p>CHAR(1): Y / N (Default = N).</p> <p>Valid for Command: add, audit, change, show, sync</p> <p>Default Value: N</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>
SIP_TIMER_PROFILE_ID	<p>Description: Foreign key: Softswitch Trunk Group Profile table. Specifies the Timer Profile ID for the Softswitch Trunk Group Profile.</p> <p>VARCHAR(16): 1-16 ASCII characters.</p> <p>Valid for Command: add, change, show, audit, sync</p> <p>Possible Value: [1_16]</p> <p>Parser: TextParser</p>
SIPT_ISUP_BASE	<p>Description: Not configurable. Mandatory if use-sipt-isup-base=Y. The SIP-T ISUP base version. This field is populated from the SIPT ISUP Version Base table.</p> <p>VARCHAR(32): 1-32 ASCII characters</p> <p>Valid for Command: show, audit, sync</p> <p>Possible Value: [1_32]</p> <p>Parser: TextParser</p>

SIPT_ISUP_VER	<p>Description: Mandatory if protocol-type=SIP-T. Defines the SIP-T or SIP-GTD version. Used only if protocol-type=SIP-T. Defined in the SIPT ISUP Version Base table. This token is only used when the protocol-type=SIPT-T. Only the GR317 version of SIP-T is supported. If the value defined in the SIPT ISUP Version Base table has a base value of sip-gtd, then the version is a SIP-GTD type. Otherwise, the version is a SIP-T type.</p> <p>VARCHAR(32): 1-32 ASCII characters. Permitted value is: GR317.</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Possible Value: [1_32]</p> <p>Parser: TextNoCaseParser</p>
START_ROW	<p>Description: Specifies to begin displaying data on the screen at a specific row.</p> <p>INTEGER: 1-100000000 (Default = 1).</p> <p>Valid for Command: show</p> <p>Default Value: 1</p> <p>Possible Value: [1_100000000]</p> <p>Parser: DecimalParser</p>
TARGET	<p>Description: Specifies the network element to receive the request.</p> <p>VARCHAR(5): 1-5 ASCII characters. Permitted values are:</p> <p>CA—Network identifier of a Call Agent.</p> <p>FSPTC (POTS/Tandem/Centrex Feature Server)—Network identifier of a specific Feature Server.</p> <p>FSAIN (AIN Feature Server)—Network identifier of AIN Feature Servers.</p> <p>Valid for Command: sync</p> <p>Mandatory: sync</p> <p>Possible Value: [1_10]</p> <p>Parser: TextParser</p>
TRUNK_SUB_GRP_TYPE	<p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: NONE</p> <p>Possible Value: NONE, BGID, TGID</p> <p>Parser: TextParser</p>

USE_PAH_HDR_FOR_ANI	<p>Description: Controls the p-asserted-id (PAI) header used to send and receive calling party information.</p> <p>CHAR(1): Y/N (Default = N).</p> <p>Y—Calling party information is derived exclusively from the PAI header on inbound calls. For outbound calls, a PAI header is sent with the calling party information if provided.</p> <p>N—Calling party information is sent or received using the From:header.</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: N</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>
USE_SIPT_ISUP_BASE	<p>Description: Mandatory if the protocol-type is SIP_T and the sipt-isup-ver is a SIP-T type. If the version selected is a GTD type, this flag is ignored. GTD does not use the base parameter. Specifies whether the SIP-T ISUP base version is included in the MIME header of the SIP-T message.</p> <p>CHAR(1): Y/N (Default = Y).</p> <p>Y—SIP-T ISUP base version is included in the MIME header of the SIP-T message.</p> <p>N—SIP-T ISUP base version is not included in the MIME header of the SIP-T message.</p> <p>Valid for Command: add, change, , audit, sync, show</p> <p>Default Value: Y</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>
VOICE_MAIL_TRUNK_GRP	<p>Description: Specifies whether the Softswitch trunk group is used for the voice-mail application.</p> <p>CHAR(1): Y/N (Default = N).</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Default Value: N</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>

North American Numbering Plan Administration State Table

The NANP-STATE table is used to define the states that are INTL-WZ1 states. The North American Numbering Plan Administration (NANPA) State (nanpa-state) table.



Note

As of Release 5.0, the original call type is preserved in billing when the routing number is changed using the Policy NXX table.

Table Name: NANP_STATE

Table Containment Area: EMS, CA

Command Types

add, audit, change, delete, help, show, sync



Caution

The **sync** command is a restricted command and is intended for repairing data only. Improper use may corrupt the database and disrupt call processing. Use with caution.

Examples

```
add nanp-state state-abbr=ak; state=alaska; country=usa;
change nanp-state state-abbr=ak; intl-wz1=y;
show nanp-state;
```

Usage Guidelines

Primary Key Token(s): STATE_ABBR

Syntax Description

AUTO_REFRESH	<p>Description: Specifies whether to display cached data on the screen.</p> <p>CHAR(1): Y/N (Default = Y).</p> <p>Y—Queries the database for the most current data.</p> <p>N—Queries the database for the most current data only if the cached data is unavailable.</p> <p>Valid for Command: show</p> <p>Default Value: Y</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>
COUNTRY	<p>Description: Country. See Table 3-1 for valid state and country combinations.</p> <p>VARCHAR(32): 1-32 ASCII characters.</p> <p>Valid for Command: add, change, show</p> <p>Possible Value: [1_32]</p> <p>Parser: TextNoCaseParser</p>

DISPLAY	<p>Description: Specifies what token information to display on the screen.</p> <p>VARCHAR(1024): 1-1024 (Default = all tokens are displayed). Permitted values are any valid token that can be shown for this command. Multiple tokens can be entered by separating with a comma.</p> <p>Valid for Command: show</p> <p>Possible Value: [1_1024]</p> <p>Parser: TextParser</p>
INTL_WZ1	<p>Description: Specifies the World Zone 1 location. See Table 3-1 for valid state and country combinations.</p> <p>CHAR(1): Y/N (Default = N).</p> <p>Valid for Command: add, audit, change, show, sync</p> <p>Default Value: N</p> <p>Possible Value: N, Y</p> <p>Parser: BooleanParser</p>
LIMIT	<p>Description: Specifies the number of rows to display on the screen.</p> <p>INTEGER: 1-100000000 (Default = 100000000).</p> <p>Valid for Command: show</p> <p>Default Value: 100000000</p> <p>Possible Value: [1_100000000]</p> <p>Parser: DecimalParser</p>
ORDER	<p>Description: Specifies whether to display data on the screen in a sorted order.</p> <p>VARCHAR(1024): 1-1024 (Default = all rows are displayed). Permitted values are any valid token that can be shown for this command. Multiple tokens can be entered by separating with a comma.</p> <p>Valid for Command: show</p> <p>Possible Value: [1_1024]</p> <p>Parser: TextParser</p>
START_ROW	<p>Description: Specifies to begin displaying data on the screen at a specific row.</p> <p>INTEGER: 1-100000000 (Default = 1).</p> <p>Valid for Command: show</p> <p>Default Value: 1</p> <p>Possible Value: [1_100000000]</p> <p>Parser: DecimalParser</p>

STATE	Description: State. See Table 3-1 for valid state and country combinations. VARCHAR(32): 1-32 ASCII characters. Valid for Command: add, change, show Possible Value: [1_32] Parser: TextNoCaseParser
STATE_ABBR	Description: Primary key. State abbreviation. See Table 3-1 for valid state and country combinations. CHAR(2): 1-2 ASCII characters. Valid for Command: add, change, delete, audit, sync, show Mandatory: add, change, delete Possible Value: [2_2] Parser: TextNoCaseParser

[Table 3-1](#) lists the valid state and country combinations for the North American Numbering Plan Administration State table.

Table 3-1 Valid State and Country Combinations

STATE-ABBR	State	Country	INTL-WZ1
AL	Alabama	USA	N
AK	Alaska	USA	Y
AZ	Arizona	USA	N
AR	Arkansas	USA	N
CA	California	USA	N
CO	Colorado	USA	N
CT	Connecticut	USA	N
DE	Delaware	USA	N
DC	District of Columbia	USA	N
FL	Florida	USA	N
GA	Georgia	USA	N
HI	Hawaii	USA	Y
ID	Idaho	USA	N
IL	Illinois	USA	N
IN	Indiana	USA	N
IA	Iowa	USA	N
KS	Kansas	USA	N
KY	Kentucky	USA	N
LA	Louisiana	USA	N
ME	Maine	USA	N
MD	Maryland	USA	N

Table 3-1 Valid State and Country Combinations (continued)

STATE-ABBR	State	Country	INTL-WZ1
MA	Massachusetts	USA	N
MI	Michigan	USA	N
MN	Minnesota	USA	N
MS	Mississippi	USA	N
MO	Missouri	USA	N
MT	Montana	USA	N
NE	Nebraska	USA	N
NV	Nevada	USA	N
NH	New Hampshire	USA	N
NJ	New Jersey	USA	N
NM	New Mexico	USA	N
NY	New York	USA	N
NC	North Carolina	USA	N
ND	North Dakota	USA	N
OH	Ohio	USA	N
OK	Oklahoma	USA	N
OR	Oregon	USA	N
PA	Pennsylvania	USA	N
RI	Rhode Island	USA	N
SC	South Carolina	USA	N
SD	South Dakota	USA	N
TN	Tennessee	USA	N
TX	Texas	USA	N
UT	Utah	USA	N
VT	Vermont	USA	N
VA	Virginia	USA	N
WA	Washington	USA	N
WV	West Virginia	USA	N
WI	Wisconsin	USA	N
WY	Wyoming	USA	N
AB	Alberta	CANADA	Y
BC	British Columbia	CANADA	Y
MB	Manitoba	CANADA	Y
NB	New Brunswick	CANADA	Y
NF	Newfoundland	CANADA	Y
NT	Northwest Territory	CANADA	Y

Table 3-1 Valid State and Country Combinations (continued)

STATE-ABBR	State	Country	INTL-WZ1
NS	Nova Scotia	CANADA	Y
VU	Nunavut Territory	CANADA	Y
ON	Ontario	CANADA	Y
PE	Prince Edward Island	CANADA	Y
PQ	Quebec	CANADA	Y
SK	Saskatchewan	CANADA	Y
YT	Yukon Territory	CANADA	Y
MX	Mexico	Mexico	Y
AS	American Samoa	American Samoa	Y
AI	Anguilla	Anguilla	Y
AN	Antigua	Antigua	Y
BA	Bahamas	Bahamas	Y
BD	Barbados	Barbados	Y
BM	Bermuda	Bermuda	Y
BV	British Virgin Islands	British Virgin Islands	Y
CQ	Cayman Islands	Cayman Islands	Y
NN	CNMI (N. Marianas)	CNMI (N. Marianas)	Y
DM	Dominica	Dominica	Y
DR	Dominican Republic	Dominican Republic	Y
GN	Grenada	Grenada	Y
GU	Guam	Guam	Y
JM	Jamaica	Jamaica	Y
RT	Montserrat	Montserrat	Y
PR	Puerto Rico	Puerto Rico	Y
KA	St. Kitts & Nevis	St. Kitts & Nevis	Y
SA	St. Lucia	St. Lucia	Y
ZF	St. Vincent	St. Vincent	Y
TR	Trinidad & Tobago	Trinidad & Tobago	Y
TC	Turks & Caicos	Turks & Caicos	Y
VI	US Virgin Islands	US Virgin Islands	Y

LERG6 Table

The LERG6 table is used to define the LERG6 tokens and values. For complete LERG6 table details, refer to the [“LERG6 Table” section on page 3-8](#).

Policy Circuit Code Table

The Policy Circuit Code table is used to define the Policy_Circuit_Code tokens and values.

Table Name: POLICY_CIRCUIT_CODE

Containment Area: EMS, CA

Command Types

add, audit, change, delete, help, show, sync



Caution

The **sync** command is a restricted command and is intended for repairing data only. Improper use may corrupt the database and disrupt call processing. Use with caution.

Usage Guidelines

Primary Key Token(s): ID, CC

Syntax Description

AUTO_REFRESH	<p>Description: Specifies whether to display cached data on the screen.</p> <p>CHAR(1): Y/N (Default = Y).</p> <p>Y—Queries the database for the most current data.</p> <p>N—Queries the database for the most current data only if the cached data is unavailable.</p> <p>Valid for Command: show</p> <p>Default Value: Y</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>
CC	<p>Description: Primary key. Country code digits. Country code as defined in ITU-T Recommendation E.164. Service provider must determine and enter accordingly. This information is often found in the front of some telephone directories. See Recommendation E.164.</p> <p>VARCHAR(5): 1-5 numeric characters.</p> <p>Valid for Command: add, audit, change, delete, show, sync</p> <p>Mandatory: add, change, delete</p> <p>Possible Value: [0_15]</p> <p>Parser: DecimalParser</p>
DISPLAY	<p>Description: Specifies what token information to display on the screen.</p> <p>VARCHAR(1024): 1-1024 (Default = all tokens are displayed). Permitted values are any valid token that can be shown for this command. Multiple tokens can be entered by separating with a comma.</p> <p>Valid for Command: show</p> <p>Possible Value: [1_1024]</p> <p>Parser: TextParser</p>

ID	<p>Description: Primary key. Foreign key: International Dial Plan Profile table. International dial plan profile ID.</p> <p>VARCHAR(16): 1-16 ASCII characters.</p> <p>Valid for Command: add, audit, change, delete, show, sync</p> <p>Mandatory: add, change, delete</p> <p>Possible Value: [1_16]</p> <p>Parser: TextParser</p>
LIMIT	<p>Description: Specifies the number of rows to display on the screen.</p> <p>INTEGER: 1-100000000 (Default = 100000000).</p> <p>Valid for Command: show</p> <p>Default Value: 100000000</p> <p>Possible Value: [1_100000000]</p> <p>Parser: DecimalParser</p>
ORDER	<p>Description: Specifies whether to display data on the screen in a sorted order.</p> <p>VARCHAR(51200): 1-51200 (Default = all rows are displayed). Permitted values are any valid token that can be shown for this command. Multiple tokens can be entered by separating with a comma.</p> <p>Valid for Command: show</p> <p>Possible Value: [1_1024]</p> <p>Parser: TextParser</p>
POLICY_ID	<p>Description: ID of the Policy or Route table that matches the policy type. Indexes the ID to the type.</p> <p>VARCHAR(16): 1-16 ASCII characters.</p> <p>Valid for Command: add, audit, change, show, sync</p> <p>Mandatory: add</p> <p>Possible Value: [1_16]</p> <p>Parser: TextParser</p>

POLICY_TYPE	<p>Description: Points to the next policy type table to be used in the sequence. Policy routing continues until policy-type=route or policy-nxx is reached. All policy-types except route point to the Policy-\$type table where \$type = odr tod percent prefix oli pop nxx. If policy-type = route, the Route table is used for routing. The policy-id indexes the Policy or Route table, whatever the case may be. Examples: If policy-type=tod, then the policy-tod table is indexed with policy-id. If policy-type=route, then the route table is indexed with policy-id.</p> <p>VARCHAR(7): 1-7 ASCII characters. Permitted values are:</p> <p>CC—Circuit Code based routing.</p> <p>CTYPE—Call Type based routing.</p> <p>NXX—Use translated DN.</p> <p>ODR—Origin Dependent Routing.</p> <p>OLI—Originating line information.</p> <p>PERCENT—Percentage based routing.</p> <p>POP—Point of presence.</p> <p>PREFIX—Prefix-based routing.</p> <p>REGION—Region-based routing.</p> <p>ROUTE—Go to Route table.</p> <p>TOD—Time-of-day routing.</p> <p>Valid for Command: add, audit, change, show, sync</p> <p>Mandatory: add</p> <p>Possible Value: CTYPE, NXX, ODR, OLI, POP, PERCENT, PREFIX, REGION, ROUTE, TOD</p> <p>Parser: TextParser</p>
-------------	---

START_ROW	<p>Description: Specifies to begin displaying data on the screen at a specific row.</p> <p>INTEGER: 1-100000000 (Default = 1).</p> <p>Valid for Command: show</p> <p>Default Value: 1</p> <p>Possible Value: [1_100000000]</p> <p>Parser: DecimalParser</p>
TYPE	<p>Description: Primary key. Points to the next policy type table to use in the sequence. Policy routing continues until policy-type=route or policy-nxx is reached. All policy-types except Route point to the Policy Type table where type = odr tod percent prefix oli pop nxx. If policy-type = route, the Route table is used for routing. The policy-id is used to index to either the Policy or Route table. Some examples are: If policy-type=tod, then the Policy Time of Day table is indexed with the policy-id. If policy-type=route, then the Route table is indexed with the policy-id.</p> <p>VARCHAR(7): 1-7 ASCII characters. Permitted values are:</p> <p>CC—Circuit Code based routing.</p> <p>CTYPE—Call type based routing.</p> <p>NXX—Use translated DN.</p> <p>ODR—Origin dependent routing.</p> <p>OLI—Originating line information.</p> <p>POP—Point of presence.</p> <p>PERCENT—Percentage based routing.</p> <p>PREFIX—Prefix-based routing.</p> <p>REGION—Region based routing.</p> <p>ROUTE (NOT PROVISIONABLE)—Go to the Route table.</p> <p>TOD—Time of day routing.</p> <p>Valid for Command: show</p> <p>Possible Value: CC</p> <p>Parser: TextParser</p>

Policy Profile Table

The Policy-Profile Table is used to define ids for Policy Routing. This table also defines the default policy to apply if the next policy to be applied in a sequence is not returned by the Policy Routing table.

Table Name: POLICY_PROFILE

Containment Area: EMS, CA, FSAIN

Command Types

add, audit, change, delete, help, show, sync



Caution

The **sync** command is a restricted command and is intended for repairing data only. Improper use may corrupt the database and disrupt call processing. Use with caution.

Examples

```
show policy-profile id=carrier0288; type=cc;
add policy-profile id=carrier0288; type=cc; default-policy-type=route;
default-policy-id=rt1;
change policy-profile id=carrier0288; type=cc; default-policy-type=route;
default-policy-id=default-rt;
delete policy-profile id=carrier0288; type=cc;
```

Usage Guidelines

Primary Key Token(s): ID, TYPE

Check Rule:

Delete Rules: FK constraints;

Syntax Description

AUTO_REFRESH	Description: Specifies whether to display cached data on the screen. CHAR(1): Y/N (Default = Y). Y—Queries the database for the most current data. N—Queries the database for the most current data only if the cached data is unavailable. Valid for Command: show Default Value: Y Possible Value: Y, N Parser: BooleanParser
DEFAULT_POLICY_ID	Description: The id of the Policy or Route table that matches the policy type. Indexes the id to the type. VARCHAR(16): 1-16 ASCII characters. Valid for Command: add, audit, change, show, sync Possible Value: [1_16] Parser: TextParser

DEFAULT_POLICY_TYPE	<p>Description: Points to the default policy type to use if the next route is not found in the Policy table. Policy routing continues until policy-type=route or policy-nxx is reached. All policy types except Route point to the Policy Type table where type = ctype odr tod percent prefix oli pop nxx. If policy-type = route, the Route table is used for routing. The policy-id is used to index to the Policy or Route table. Some examples are:</p> <p>If policy-type=tod, then the Policy TOD table is indexed with policy-id.</p> <p>If policy-type=route, then the Route table is indexed with policy-id.</p> <p>VARCHAR(7): 1-7 ASCII characters. Permitted values are:</p> <p>CC—Circuit code based routing.</p> <p>CTYPE—Call type based routing.</p> <p>NXX—Use translated DN.</p> <p>ODR—Origin dependent routing.</p> <p>OLI—Originating line information.</p> <p>POP—Point of presence.</p> <p>PERCENT—Percentage based routing.</p> <p>PREFIX—Prefix-based Routing.</p> <p>REGION—Region based Routing</p> <p>ROUTE—Go to Route table.</p> <p>TOD—Time of day routing.</p> <p>Valid for Command: add, audit, change, show, sync</p> <p>Possible Value: CC, CTYPE, NXX, ODR, OLI, POP, PERCENT, PREFIX, REGION, ROUTE, TOD</p> <p>Parser: TextParser</p>
DESCRIPTION	<p>Description: Service provider-defined description.</p> <p>VARCHAR(64): 1-64 ASCII characters.</p> <p>Valid for Command: add, change, audit, sync, show</p> <p>Possible Value: [1_64]</p> <p>Parser: TextParser</p>
DISPLAY	<p>Description: Specifies what token information to display on the screen.</p> <p>VARCHAR(1024): 1-1024 (Default = all tokens are displayed). Permitted values are any valid token that can be shown for this command. Multiple tokens can be entered by separating with a comma.</p> <p>Valid for Command: show</p> <p>Possible Value: [1_1024]</p> <p>Parser: TextParser</p>

ID	<p>Description: Primary key. Policy id.</p> <p>VARCHAR(16): 1-16 ASCII characters.</p> <p>Valid for Command: add, audit, change, delete, show, sync</p> <p>Mandatory: add, change, delete</p> <p>Possible Value: [1_16]</p> <p>Parser: TextParser</p>
LIMIT	<p>Description: Specifies the number of rows to display on the screen.</p> <p>INTEGER: 1-100000000 (Default = 100000000).</p> <p>Valid for Command: show</p> <p>Default Value: 100000000</p> <p>Possible Value: [1_100000000]</p> <p>Parser: DecimalParser</p>
MASTER	<p>Valid for Command: sync</p> <p>Mandatory: sync</p> <p>Possible Value: [1_10]</p> <p>Parser: TextParser</p>
ORDER	<p>Description: Specifies whether to display data on the screen in a sorted order.</p> <p>VARCHAR(51200): 1-51200 (Default = all rows are displayed). Permitted values are any valid token that can be shown for this command. Multiple tokens can be entered by separating with a comma.</p> <p>Valid for Command: show</p> <p>Possible Value: [1_1024]</p> <p>Parser: TextParser</p>
START_ROW	<p>Description: Specifies to begin displaying data on the screen at a specific row.</p> <p>INTEGER: 1-100000000 (Default = 1).</p> <p>Valid for Command: show</p> <p>Default Value: 1</p> <p>Possible Value: [1_100000000]</p> <p>Parser: DecimalParser</p>

TARGET	<p>Description: Specifies the network element to receive the request.</p> <p>VARCHAR(5): 1-5 ASCII characters. Permitted values are:</p> <p>CA—Network identifier of a Call Agent.</p> <p>FSPTC (POTS/Tandem/Centrex Feature Server)—Network identifier of a specific Feature Server.</p> <p>FSAIN (AIN Feature Server)—Network identifier of AIN Feature Servers.</p> <p>Valid for Command: sync</p> <p>Mandatory: sync</p> <p>Possible Value: [1_10]</p> <p>Parser: TextParser</p>
TYPE	<p>Description: Primary key. Points to the next policy type table to use in the sequence. Policy routing continues until policy-type=route or policy-nxx is reached. All policy-types except Route point to the Policy Type table where type = odr tod percent prefix oli pop nxx. If policy-type = route, the Route table is used for routing. The policy-id is used to index to either the Policy or Route table. Some examples are:</p> <p>If policy-type=tod, then the Policy Time of Day table is indexed with the policy-id.</p> <p>If policy-type=route, then the Route table is indexed with the policy-id.</p> <p>VARCHAR(7): 1-7 ASCII characters. Permitted values are:</p> <p>CC—Circuit Code based routing.</p> <p>CTYPE—Call type based routing.</p> <p>NXX—Use translated DN.</p> <p>ODR—Origin dependent routing.</p> <p>OLI—Originating line information.</p> <p>POP—Point of presence.</p> <p>PERCENT—Percentage based routing.</p> <p>PREFIX—Prefix-based routing.</p> <p>REGION—Region based routing.</p> <p>ROUTE (NOT PROVISIONABLE)—Go to the Route table.</p> <p>TOD—Time of day routing.</p> <p>Valid for Command: add, audit, change, delete, show, sync</p> <p>Mandatory: add, change, delete</p> <p>Possible Value: CC, CTYPE, NXX, ODR, OLI, POP, PERCENT, PREFIX, REGION, ROUTE, TOD</p> <p>Parser: TextParser</p>

Policy Call Type Table

The Policy Call Type table is used to define the Policy Call Type tokens and values.

Table Name: POLICY_CALL_TYPE

Containment Area: EMS, CA

Command Types add, audit, change, delete, help, show, sync



Caution

The **sync** command is a restricted command and is intended for repairing data only. Improper use may corrupt the database and disrupt call processing. Use with caution.

Usage Guidelines Primary Key Token(s): ID, CALL_TYPE

Syntax Description	AUTO_REFRESH	<p>Description: Specifies whether to display cached data on the screen.</p> <p>CHAR(1): Y/N (Default = Y).</p> <p>Y—Queries the database for the most current data.</p> <p>N—Queries the database for the most current data only if the cached data is unavailable.</p> <p>Valid for Command: show</p> <p>Default Value: Y</p> <p>Possible Value: Y, N</p> <p>Parser: BooleanParser</p>
	CALL_TYPE	<p>Description: Call type based on the dialed number.</p> <p>VARCHAR(9): 1-9 ASCII characters. Permitted values are:</p> <p>DA—NPA-555-1212 calls.</p> <p>DA-TOLL—1+NPA-555-1212 calls.</p> <p>976—Information services calls.</p> <p>INFO—Information services calls.</p> <p>TW—Time and temperature service.</p> <p>Valid for Command: add, audit, change, delete, show, sync</p> <p>Mandatory: add, change, delete</p> <p>Possible Value: [1_16]</p> <p>Parser: TextNoCaseParser</p>

DISPLAY	<p>Description: Specifies what token information to display on the screen.</p> <p>VARCHAR(1024): 1-1024 (Default = all tokens are displayed). Permitted values are any valid token that can be shown for this command. Multiple tokens can be entered by separating with a comma.</p> <p>Valid for Command: show</p> <p>Possible Value: [1_1024]</p> <p>Parser: TextParser</p>
ID	<p>Description: Primary key. Unique call control route identifier.</p> <p>VARCHAR(16): 1-16 ASCII characters.</p> <p>Valid for Command: add, audit, change, delete, show, sync</p> <p>Mandatory: add, change, delete</p> <p>Possible Value: [1_16]</p> <p>Parser: TextParser</p>
LIMIT	<p>Description: Specifies the number of rows to display on the screen.</p> <p>INTEGER: 1-100000000 (Default = 100000000).</p> <p>Valid for Command: show</p> <p>Default Value: 100000000</p> <p>Possible Value: [1_100000000]</p> <p>Parser: DecimalParser</p>
ORDER	<p>Description: Specifies whether to display data on the screen in a sorted order.</p> <p>VARCHAR(51200): 1-51200 (Default = all rows are displayed). Permitted values are any valid token that can be shown for this command. Multiple tokens can be entered by separating with a comma.</p> <p>Valid for Command: show</p> <p>Possible Value: [1_1024]</p> <p>Parser: TextParser</p>
POLICY_ID	<p>Description: ID of the Policy or Route table that matches the policy type. Indexes the ID to the type.</p> <p>VARCHAR(16): 1-16 ASCII characters.</p> <p>Valid for Command: add, audit, change, show, sync</p> <p>Mandatory: add</p> <p>Possible Value: [1_16]</p> <p>Parser: TextParser</p>

POLICY_TYPE	<p>Description: Points to the next policy type table to use in the sequence. Policy routing continues until policy-type=route or policy-nxx is reached. All policy-types except route point to the Policy-\$type table where \$type = odr tod percent prefix oli pop nxx. If policy-type = route, the Route table is used for routing. The policy-id indexes the Policy or Route table, whatever the case may be.</p> <p>VARCHAR(7): 1-7 ASCII characters. Permitted values are:</p> <p>CC—Circuit Code based routing</p> <p>CTYPE—Call Type based routing</p> <p>NXX—Use translated DN.</p> <p>ODR—Origin Dependent Routing.</p> <p>OLI—Originating line information.</p> <p>PERCENT—Percentage based routing</p> <p>POP—Point of presence.</p> <p>PREFIX—Prefix-based routing.</p> <p>REGION—Region-based routing.</p> <p>ROUTE—Go to Route table.</p> <p>TOD—Time-of-day routing</p> <p>Valid for Command: add, audit, change, show, sync</p> <p>Mandatory: add</p> <p>Possible Value: CC, NXX, ODR, OLI, POP, PERCENT, PREFIX, REGION, ROUTE, TOD</p> <p>Parser: TextParser</p>
-------------	---

START_ROW	<p>Description: Specifies to begin displaying data on the screen at a specific row.</p> <p>INTEGER: 1-100000000 (Default = 1).</p> <p>Valid for Command: show</p> <p>Default Value: 1</p> <p>Possible Value: [1_100000000]</p> <p>Parser: DecimalParser</p>
TYPE	<p>Description: Primary key. Points to the next policy type table to use in the sequence. Policy routing continues until policy-type=route or policy-nxx is reached. All policy-types except Route point to the Policy Type table where type = odr tod percent prefix oli pop nxx. If policy-type = route, the Route table is used for routing. The policy-id is used to index to either the Policy or Route table. Some examples are: If policy-type=tod, then the Policy Time of Day table is indexed with the policy-id. If policy-type=route, then the Route table is indexed with the policy-id.</p> <p>VARCHAR(7): 1-7 ASCII characters. Permitted values are:</p> <p>CC—Circuit Code based routing.</p> <p>CTYPE—Call type based routing.</p> <p>NXX—Use translated DN.</p> <p>ODR—Origin dependent routing.</p> <p>OLI—Originating line information.</p> <p>POP—Point of presence.</p> <p>PERCENT—Percentage based routing.</p> <p>PREFIX—Prefix-based routing.</p> <p>REGION—Region based routing.</p> <p>ROUTE (NOT PROVISIONABLE)—Go to the Route table.</p> <p>TOD—Time of day routing.</p> <p>Valid for Command: show</p> <p>Possible Value: CTYPE</p> <p>Parser: TextParser</p>

Dial Plan Profile Table

As indicated in the description of the SKIP-DIAL-PLAN flag, if set, the number will not be analyzed in the dial-plan. It seems as though the functionality can be achieved by not provisioning any dial-plan entries. However, within the implementation, table is looked at several times to attempt to find the longest match. To skip this process, Skip-dial-plan is used. For additional information on the Dial Plan Profile table, refer to the [“Dial Plan Profile” section on page 1-36](#).

