**C H A P T E R 1**

Digit Translations

Revised: August 10, 2011, OL-24996-01

Introduction

The Cisco BTS 10200 Softswitch digit manipulation or digit translation function enables manipulating or translating the digit string, the nature of address (NOA), or both. Digit manipulation or translation can take place at several points in call processing, as illustrated in Figure 1-1.

Figure 1-1 **Digit Manipulation Locations and Stages**
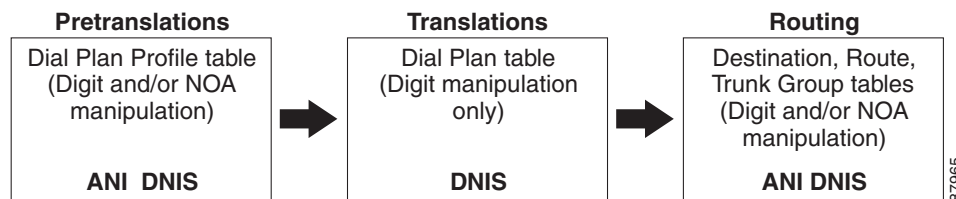
The digit and/or NOA manipulation rules can be applied during the pretranslations, translations, or routing stage. Additionally, digit manipulations or digit translations can occur at either the originating basic call system manager (OBCSM) or the terminating basic call system manager (TBCSM). The following sections describe the tables and tokens required for digit and/or NOA manipulation at each of these three stages. For details on routing and call processing including actions at specific points in call processing, refer to Chapter 2, "Routing" and to Chapter 5, "Dial Plans and Routing."

Pre-Translations—In the pre-translations stage, the dial-plan-profile table is used to specify if automatic number identification (ANI), dialed number identification service (DNIS) or both are to be manipulated. The purpose of the pre-translation stage is to normalize the digits as required during the translations stage. For example: if a region supports 7-digit dialing, you can use DNIS manipulation in the dial-plan-profile to add a home numbering plan area (HNPA) to make it a 10-digit directory number (DN).

Translations—The dial-plan table can be used to manipulate the called party number (DNIS). Simple delete and prefix functionality is supported.

Routing—The routing is performed in the analyzed info point in call (PIC). The Destination table and/or Route table can be used to specify digit manipulation of the ANI, the DNIS or both. The out-pulsing number can be normalized by use of the destination table. In addition, if special manipulation is required based on the route selected, then special manipulation can be specified for each trunk group within a route.

If the Called Party Number terminates within the Cisco BTS 10200, and the subscriber number points to a trunk group, then digit manipulation rules for the ANI, the DNIS, or both can be specified in the Trunk Group table.

If the call is an interLATA call, or requires Carrier Routing, the ANI/DNIS digit manipulation identification (ID) specified in the Destination table is ignored and carrier based routing is performed.

Figure 1-2 illustrates the provisioning relationships for the Cisco BTS 10200 dial plans, which include the ANI/DNIS digital manipulation functions.

Figure 1-2   ***Dial Plan Provisioning Relationships***
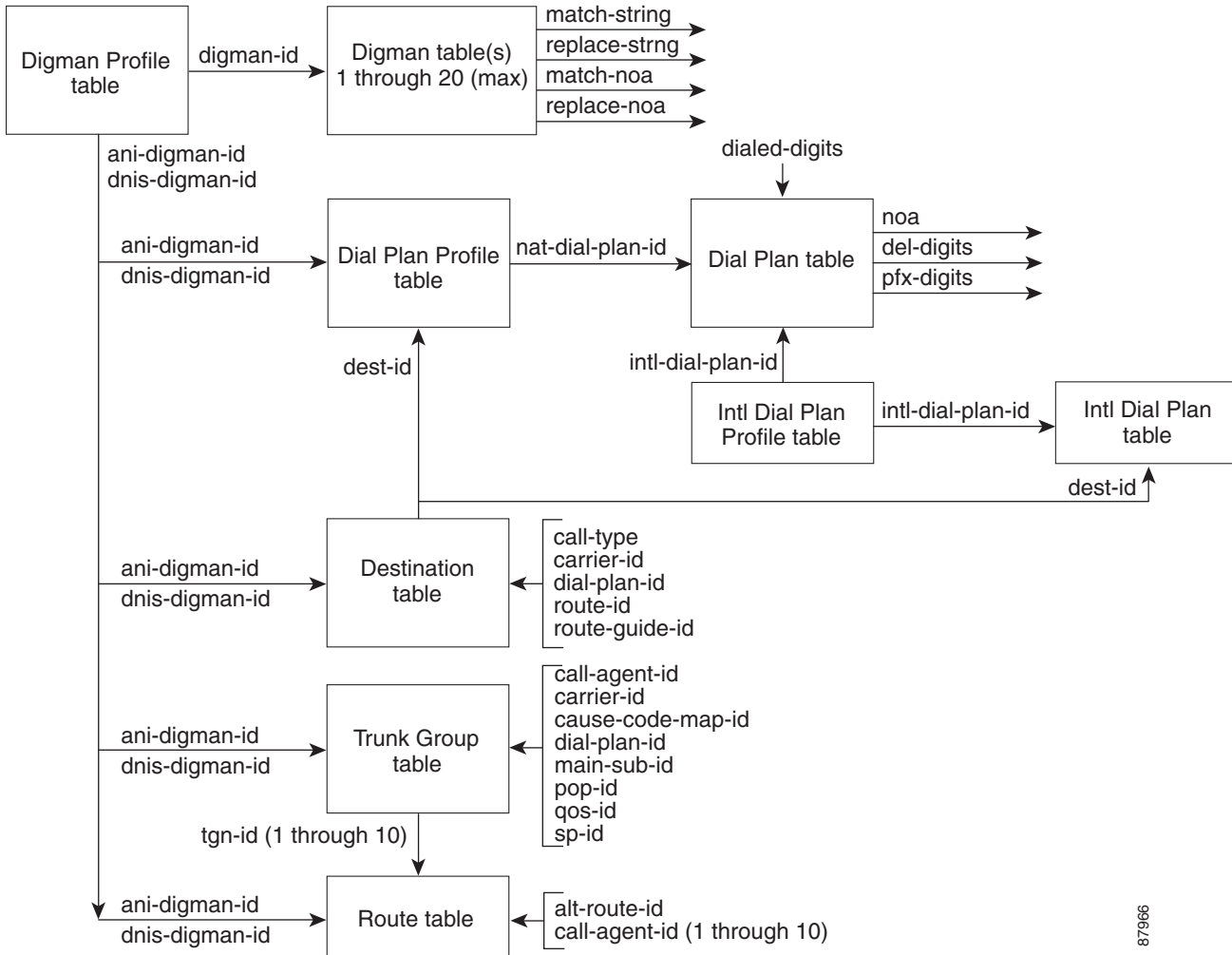
Digit Manipulation

Digit manipulation or digit translation is performed based on as many as 20 different digit manipulation (digman) tables as designated by the digman-id and rule number, each of which can have a unique set of match-string and replace-string tokens and/or match-NOA and replace-NOA tokens.

The match-string is compared to the input-string. If a match is found, based on the rules specified here, then the replace-string replaces the matched string in all further call processing actions.

The match-string and replace-string tokens are constructed using the characters specified in Table 1-1.

Table 1-1   Digit Manipulation Rules Specifications

Character(s)	Action
Digits 0 through 9, asterisk *, pound sign #	Digits found on a key-pad. Valid for use at any position in the input-string token, match-string token, and replace-string token.
Caret (^)	Beginning of String Character—Indicates a match to the beginning of the string. The caret character can only be the first character of the match-string. If the caret character is specified by itself, the input string is prefixed by the replace-string. For examples of caret character usage, refer to the "Caret (^) Character" section.
Dollar sign ($)	End of String Character—The dollar sign character can be specified as only the last character of the match string. If the dollar sign character is specified by itself, the replace-string is appended to the input-string. The dollar sign character is used to specify the end of string. If digits at the end of the string are to be matched, the match-string is terminated by the "$" character. For examples of dollar sign character usage, refer to "Dollar Sign ($) Character" section.
Dot (.)	If the dot character is specified as a leading character, beginning of the string is assumed. If the dot character is specified as the last character, end of string is assumed. The dot character is used to identify the position of the match. The digits occupying the position specified by the dot character are skipped during the match operation. For examples of dot character usage, refer to "Dot (.) Character" section.
Question Mark (?)	The question mark character can appear anywhere, but multiple question marks have to be with another question mark. The question mark character cannot appear as a single character. If only one digit is to be manipulated, then the "?" mark should be preceded by ^ (caret) or followed by a "$" (dollar) to signify the first digit or the last digit respectively. Used as a wildcard. Matches any one character. For examples of question mark character usage, refer to "Question Mark (?) Character" section. **Note** The "?" character by itself is treated as a request for help.
Percent sign (%)	Replace Character—The percent sign character can only be specified as the first character of the string. If specified by itself, the input string is replaced by the replace-string. Also used as a wildcard and when specified is used to match 0 or more characters preceding the digits specified in the match-string. For examples of percent sign character usage, refer to "Percent (%) Character (Match and Replace)" section.

Table 1-1   *Digit Manipulation Rules Specifications (continued)*

Character(s)	Action
Ampersand (&)	Prefix Character—The ampersand character can only be specified as the last character of the string. If specified by itself, it is used to indicate that no replace function is required on the matched string. The ampersand character is used to indicate if the input string matches with the match-string. If it is specified with digits preceding it, the specified digits are prefixed to the matched string. For examples of ampersand character usage, refer to "Ampersand (&) Character" section.
None (or none)	Can only be specified by itself.
	The word "none" is used to specify null string. Can be used for both match-string and replace-string.
	Example 1:
	`match-string=none; replace-string=4692550000; indicates that when input string is null, replace it with 4692550000.`
	Example 2:
	`match-string=469255; replace-string=none; indicates that if a match is found, replace the matched string with null. If input string is 4692551234, after the digit manipulation, the resultant string will be 1234.`

Match-string rules:

- Consist of digits 0-9, *, #, ^, $, dot(.), question mark (?), percent (%) or word "none".

- Caret (^) if specified, can only be first character of the string.

- Dollar Sign ($) if specified, can only be the last character of the string.

- Percent (%) if specified, can only be first character of the string.

- Phrase "none" or "NONE" can only appear by itself.

The following rules are not enforced by the parser:

- Question Mark(s) (?) if specified, cannot appear in between a digit string (example: match-string=12??56; is invalid, but match-string=12????; is valid).

- Dots (.) if specified, can appear as leading dots; trailing dots; or both. (Examples: match-string=...555; match-string=555....; match-string=...555....;)

Replace-string rules:

- Consist of digits 0-9, *, #, ampersand (&) or word "none".

- Ampersand (&) if specified, can appear by itself or be the last character of the string.

- Word "none" or "NONE" can only appear by itself.

Character Function

The following subsections provide information on the Cisco BTS 10200 character function.

Caret (^) Character

The caret (^) character is used to specify beginning of a string. The caret character is used when specific leading digits are to be deleted or replaced. The caret character is also used when digits are to be prefixed to the input string.

The following examples specify the use of the caret character:

Example 1

```
input-string=0119127210112; match-string=^011; replace-string=none. In this example, the
digit string 011 at the beginning of input string will be replaced with null. The
resultant output string=9127210112.
```

Example 2

```
input-string=9127210112; match-string=^; replace-string=011. In this example, since the
digit string 011 is prefixed to the input-string. The resultant
output-string=0119127210112.
```

Example 3

```
Enter 1 to continue translations 1
Enter input string 14692551234
Match string ^1
Replace string none
MATCHED

Output string=4692551234
```

Example 4

```
Enter 1 to continue translations 1
Enter input string 4692551234
Match string ^469255
Replace string 5
MATCHED

Output string=51234
```

Example 5

```
Enter 1 to continue translations 1
Enter input string 222
Match string ^
Replace string 1
MATCHED

Output string=1222
```

Dollar Sign ($) Character

The dollar sign ($) character is used to search from the end of string backwards instead of from the beginning of the string. The dollar sign character is also used when digits are to be appended to the end of the string, deleted or replaced from the end of the string.

If match-string=1234$, the last 4 digits will be 1234.

If match-string=????$, replace last 4 digits with the replace-string.

The following examples show the use of dollar sign character.

Example 1

```
input string=4692551234; match-string=????$; replace-string=0000. In this example, digits
1234 will be replaced with 0000. The resultant output string=4692550000.
```

Example 2

```
input string=469255; match-string=$; replace-string=0000. In this example, digits 0000
will be appended to the input-string. The resultant output string=4692550000.
```

Example 3

```
Enter 1 to continue translations 1
Enter input string 4692551234
Match string ????$
Replace string none
MATCHED

Output string=469255
```

Example 4

```
Enter 1 to continue translations 1
Enter input string 4692551234
Match string ????$
Replace string 0000
MATCHED

Output string=4692550000
```

Example 5

```
Enter 1 to continue translations 1
Enter input string 469255
Match string $
Replace string 0000
MATCHED

Output string=4692550000
```

Dot (.) Character

The dot (.) character is used to indicate string position at which match operation is to be performed. If a dot character is specified as the first character of the match-string, the digits specified by each dot character are skipped until a question mark or any digit is encountered.

The dot character can also be used to check the length of the input string (see examples below).

Example 1

```
match-string="...555" indicates that 555 appear in digit position 4-6. This will match on
the following input string: 4695551234.
```

Example 2

```
"555..." Indicates that find a match on 555 where there are 4 more digits that follow 555.
This will match any of the following input strings: 4695551234 or 5551234.
```

Example 3

```
Enter 1 to continue translations 1
Enter input string 4695551234
Match string ...555
Replace string &
MATCHED

Output string=4695551234
```

Example 4

```
Enter 1 to continue translations 1
Enter input string 4695551234
Match string .......... #CHECK IF INPUT STRING LENGTH=10
Replace string &
MATCHED

Output string=4695551234
```

Example 5

```
Enter 1 to continue translations 1
Enter input string 14695551234
Match string ..........
Replace string &
Not MATCHED

Output string=14695551234
```

Example 6

```
Enter 1 to continue translations 1
Enter input string 222333
Match string ...33
Replace string none
MATCHED

Output string=2223
```

Example 7

```
Enter 1 to continue translations 1
Enter input string 22233
Match string ^...
Replace string none
Not MATCHED

Output string=22233
```

Question Mark (?) Character

The question mark (?) character is used to specify a wild card character based on its position in the match-string. Each question mark character represents one digit character.

If a question mark is specified as a leading character in a match-string, the beginning of input string is assumed. Match-string "???555" indicates that you should look for any three digits followed by digits 555. The question mark character can not be specified as the only character as it conflicts with the use of "?" as a help character. So, if the first digit is to be matched, use ^? to represent first digit and ?$ to represent the last digit of a digit-string.

The following examples show the use of question mark (?) character.

Example 1

```
"???555" indicates that 555 appear in digit position 4-6. This will match the following
input string: 4695551234.
```

Example 2

```
"555????" Indicates that find a match on 555 where there are 4 more digits that follow
555. This will match any of the following input strings: 4695551234 or 5551234. Digits
5551234 will be replaced with the replace string.
```

Example 3

```
"^?" indicates to match on the 1st digit
```

Example 4

```
"?$" indicates to match on the last digit
```

Example 5

```
Enter 1 to continue translations 1
Enter input string 14692551234
Match string ^1???
Replace string none
MATCHED

Output string=2551234
```

Example 6

```
Enter 1 to continue translations 1
Enter input string 4692551234
Match string ????$
Replace string none
MATCHED

Output string=469255
```

Example 7

```
Enter 1 to continue translations 1
Enter input string 4695551234
Match string ???555
Replace string 5
MATCHED

Output string=51234
```

Example 8

```
Enter 1 to continue translations 1
Enter input string 4692551234
Match string ???555
Replace string none
Not MATCHED

Output string=4692551234
```

Percent (%) Character (Match and Replace)

The percent character (%) is used to specify the match and replace function. That is, if a match occurs, replace the matched string with the replace-string. The percent character is also used as a wild card character.

If match-string=%, the whole input-string is replaced by the replace-string.

If match-string=%nn, the input-string is searched from the beginning until it matches pattern nn, and the whole string from the beginning to the end of pattern is replaced with the replace-string.

If match-string=%...nnn, the input-string is searched for pattern nnn in digit positions 4, 5, and 6. If a match occurs, the digit string from the beginning including the pattern nnn is replaced with the replace-string.

The following examples show the use of the percent (%) character.

Example 1

```
%555 - indicates match on 555 and any number of digits preceding it. This will match on of
the following input string: 4695551234. Digits 469555 will be replaced with the replace
string.
```

Example 2

```
% - indicates match on any number of digits.
```

Example 3

```
Enter 1 to continue translations 1
Enter input string 4692551234
Match string %
Replace string 55555
MATCHED

Output string=55555
```

Example 4

```
Enter 1 to continue translations 1
Enter input string 4692551234
Match string %255
Replace string 5
MATCHED

Output string=51234
```

Example 5

```
Enter 1 to continue translations 1
Enter input string 4692551234
Match string %...255MATCH DIGITS 255 IN DIGIT POSITIONS 4,5,6
Replace string 5
MATCHED

Output string=51234
```

Example 6

```
Enter 1 to continue translations 1
Enter input string 4692551234
Match string %...255?MATCH DIGITS 255 IN DIGIT POSITIONS 4,5,6 AND 1 MORE DIGIT
Replace string 55
MATCHED

Output string=55234
```

Ampersand (&) Character

The ampersand character (&) is used in the replace-string to leave the matched string as is without replacing it with the replace string. If ampersand is specified by itself, the digit manipulation leaves the input string as is. If a digit string is specified followed by an ampersand, the digit string is prefixed to the matched string.

The following examples show the use of the ampersand (&) character.

Example 1

```
input-string=4695551212; match-string=555.; replace-string=&; will return an indication of
match, but the output string will be unaffected.
```

Example 2

```
input string=4695551234; match-string=???555.; replace-string=1&; In this example, if the
input string is 10 digits long, and digits(4-6) 555, prefix the input string with 1. The
resultant output string=14695551234.
```

Example 3

```
Enter 1 to continue translations 1
Enter input string 4692551234
Match string ???255
Replace string 1&
MATCHED

Output string=14692551234
```

Example 4

```
Enter 1 to continue translations 1
Enter input string 14692551234
Match string ^1..........
Replace string &
MATCHED

Output string=14692551234
```

Example 5

```
Enter input string 4695551212
Match string ^...555....
Replace string &
MATCHED

Output string=4695551212
```

Example 6

```
Enter 1 to continue translations 1
Enter input string *85#
Match string *
Replace string &
MATCHED

Output string=*85#
```

Example 7

```
Enter 1 to continue translations 1
Enter input string *85#
Match string #
Replace string &
MATCHED

Output string=*85#
```

Delete Digits Function

The Cisco BTS 10200 digit manipulation function supports the following delete capabilities:

- Deletion of leading digits

- Deletion of specific leading digits

- Deletion of trailing digits

- Deletion of specific trailing digits

- Deletion of leading digits only if the length matches

- Deletion of trailing digits only if the length matches

Deletion of Leading Digits

The Cisco BTS 10200 digit manipulation functions supports the deletion of leading digits, as indicated in the following example:

```
Add digman id=del3; match-string=???; replace-string=none; or
Add digman id=del3; match-string=^???; replace-string=none;
**************************************************
Enter input string 4692551234
Match string ???
Replace string none
MATCHED

Output string=2551234
**************************************************
```

Deletion of Specific Leading Digits

The Cisco BTS 10200 digit manipulation function supports the deletion of specific leading digits, as indicated in the following example:

```
Add digman id=del00; match-string=^00; replace-string=none;
**************************************************
Enter input string 0012345
Match string ^00
Replace string none
MATCHED

Output string=12345
**************************************************
```

Deletion of Trailing Digits

The Cisco BTS 10200 digit manipulation function supports the deletion of trailing digits, as indicated in the following example:

```
Add digman id=dellast4; match-string=????$; replace-string=none;
***************************************************
Enter input string 4692551234
Match string ????$
Replace string none
MATCHED

Output string=469255
***************************************************
```

Deletion of Specific Trailing Digits

The Cisco BTS 10200 digit manipulation function supports deletion of specific trailing digits, as indicated in the following example:

```
Add digman id=delx1212; match-string=1212$; replace-string=none;
***************************************************
Enter input string 4695551212
Match string 1212$
Replace string none
MATCHED

Output string=469555
***************************************************
```

Deletion of Leading Digits Only If the Length Matches

The Cisco BTS 10200 digit manipulation function supports the deletion of leading digits only if the length of input-string matches the length implied by the match-string, as indicated in the following example:

```
Add digman id=del310; match-string=^???.......; replace-string=none; or
Add digman id=del310; match-string=???.......; replace-string=none;
***************************************************
Enter input string 4692551234
Match string ???.......
Replace string none
MATCHED

Output string=2551234
***************************************************
```

Deletion of Trailing Digits Only If the Length Matches

The Cisco BTS 10200 digit manipulation function supports deletion of trailing digits only if the length of input-string matches the length implied by the match-string, as indicated in the following example:

```
Add digman id=dellast4; match-string=......????; replace-string=none; or
Add digman id=del310; match-string=^......????$; replace-string=none;
***************************************************
Enter input string 4692551234
Match string ......????
Replace string none
MATCHED

Output string=469255
***************************************************
```

Prefix Digits Function

The Cisco BTS 10200 digit manipulation function supports the prefix of digits. The following prefix functions are supported:

- Prefix leading digits
- Prefixing of digits only if length matches

Prefix Leading Digits

The Cisco BTS 10200 digit manipulation function supports the prefixing of leading digits, as indicated in the following example:

```
Add digman id=pfx469; match-string=^; replace-string=469;
***************************************************
Enter input string 5551234
Match string ^
Replace string 469
MATCHED

Output string=4695551234
***************************************************
```

Prefixing of Digits Only If Length Matches

The Cisco BTS 10200 digit manipulation function supports the prefixing of digits only if length matches, as indicated in the following example:

```
Add digman id=pfx469if7; match-string=^.......; replace-string=469; or
Add digman id=pfx469if7; match-string=.......; replace-string=469;
***************************************************
Enter input string 5551234
Match string ^.......
Replace string 469
MATCHED

Output string=4695551234
***************************************************
```

Replace (Delete and Prefix) Digits Function

The Cisco BTS 10200 digit manipulation function supports the digit replacement based on matched string; that is, the matched string will be replaced with the replace-string.

- Replacement of leading digits

- Replacement of trailing digits

- Replacement of specific digits

- Replacement of leading digits only if length matches

- Replacement of specific digits in a specific digit position

- Replacement of digits in a specific digit position

- Replacement of digits only if null

- Replacement of any input string

Replacement of Leading Digits

The Cisco BTS 10200 digit manipulation function supports the replacement (delete and prefix) of leading digits, as indicated in the following examples:

```
Add digman id=del6pfx5; match-string=??????; replace-string=5; or
Add digman id=del6pfx5; match-string=^??????; replace-string=5;
**************************************************
Enter input string 4692551234
Match string ??????
Replace string 5
MATCHED

Output string=51234
**************************************************
Enter input string 4692551234
Match string ^??????
Replace string 5
MATCHED

Output string=51234
**************************************************
```

Replacement of Trailing Digits

The Cisco BTS 10200 digit manipulation function supports the replacing trailing digits, as indicated in the following example:

```
Add digman id=del4rep0000; match-string=????$; replace-string=0000;
**************************************************
Enter input string 4692551234
Match string ????$
Replace string 0000
MATCHED

Output string=4692550000
**************************************************
```

Replacement of Specific Digits

The Cisco BTS 10200 digit manipulation function supports specific digits replacement only if the input string matches specific digits, as indicated in the following example:

```
Add digman id=del469255pfx5; match-string=469255; replace-string=5;
***************************************************
Enter input string 4692551234
Match string 469255
Replace string 5
MATCHED

Output string=51234
***************************************************
```

Replacement of Leading Digits Only If Length Matches

The Cisco BTS 10200 digit manipulation function supports leading digits replacement only if the length matches, as indicated in the following examples:

```
Add digman id=del6pfx5; match-string=469255....; replace-string=5;
***************************************************
Enter input string 4692551234
Match string 469255....
Replace string 5
MATCHED

Output string=51234
***************************************************
Enter input string 4692550
Match string 469255....
Replace string 5
Not MATCHED

Output string=4692550
***************************************************
```

Replacement of Specific Digits In a Specific Digit Position

The Cisco BTS 10200 digit manipulation function supports specific digits replacement specified by position, as indicated in the following example:

```
Add digman id=rep555; match-string=...555....; replace-string=222;
***************************************************
Enter input string 4695551234
Match string ...555....
Replace string 222
MATCHED

Output string=4692221234
***************************************************
```

Replacement of Digits In a Specific Digit Position

The Cisco BTS 10200 digit manipulation function supports the digits replacement in the specific digit position, as indicated in the following examples:

```
Add digman id=rep456w222; match-string=...???....; replace-string=222;
Add digman id=replast4; match-string=......????; replace-string=0000;
**************************************************
Enter input string 4695551234
Match string ...???....
Replace string 222
MATCHED

Output string=4692221234
**************************************************
Enter input string 4695551234
Match string ......????
Replace string 0000
MATCHED

Output string=4695550000
**************************************************
```

Replacement of Digits Only If Null

The Cisco BTS 10200 digit manipulation function supports digit replacement if input digit string is null, as indicated in the following examples:

```
Add digman id=repifnull; match-string=none; replace-string=4692550000;
**************************************************
Enter input string none
Match string none
Replace string 4692550000
MATCHED

Output string=4692550000
**************************************************
Enter input string 4695551234
Match string none
Replace string 4692550000
Not MATCHED

Output string=4695551234
**************************************************
```

Replacement of Any Input String

The Cisco BTS 10200 digit manipulation function supports the replacement of any input string with the replace-string, as indicated in the following examples:

```
Add digman id=replace; match-string=%; replace-string=4692550000;
**************************************************
Enter input string none
Match string %
Replace string 4692550000
MATCHED

Output string=4692550000
**************************************************
Enter input string 4695551234
Match string %
Replace string 4692550000
MATCHED

Output string=4692550000
**************************************************
```

Nature of Address Manipulation

The Cisco BTS 10200 Digit Manipulation table also supports NOA manipulation.

To manipulate the NOA, the match-string and the replace-string should be null.

```
Add digman id=dg1; rule=1; match-noa=any; replace-noa=subscriber;
```

If both digit and NOA manipulation rules are defined, the digit manipulation is only performed if the NOA value specified in the match-noa matches the input NOA and the match-string matches the input string.

```
Add digman id=dg1; rule=1; match-string=%255; replace-string=5; match-noa=national;
replace-noa=abbr;
```

Table 1-2   *Nature Of Address Table*

NOA	Description
950	NOA used to specify 950 call.
ANY	When specified matches any of the other NOAs. This NOA can only be specified in the MATCH-NOA field.
ABBR	NOA used to specify abbreviated number.
CUT-THRU	NOA used to specify no number present, cut-thru call.
INTL	NOA used to specify international number
INTL-OPR	NOA used to specify international number, operator requested (valid for called party number only).
INTL	NOA used to specify unique international number.
NATIONAL	NOA used to specify national number.
NAT-OPR	NOA used to specify national number, operator requested (valid for called party number only).
NETWORK	NOA used to specify network.
NON-UNIQUE-INTL	NOA used to specify non-unique international number (valid for calling party number).
NON-UNIQUE-NATIONAL	NOA used to specify non-unique national number (valid for calling party number).
NON-UNIQUE-SUBSCRIBER	NOA used to specify non-unique subscriber number (valid for calling party number).
NS0	NOA used to specify network specific (111 1000) number.
NS1	NOA used to specify network specific (111 1001) number.
NS2	NOA used to specify network specific (111 1010) number.
NS3	NOA used to specify network specific (111 1011) number.
NS4	NOA used to specify network specific (111 1100) number.
NS5	NOA used to specify network specific (111 1101) number.
NS6	NOA used to specify network specific (111 1110) number.
OPERATOR	NOA used to specify an operator call.
PORTED-NUMBER-WITHOUT-RN	The ported number may be prefixed with the network ID, or may not be prefixed with anything (DN or NTWK-ID+DN), but does not include the switch ID.
PORTED-NUMBER-WITH-RN	The ported Number is prefixed with the network ID and switch ID (RN+DN).

Table 1-2   Nature Of Address Table (continued)

NOA	Description
PRIVATE	NOA used to specify private numbering plan.
RESERVED	Reserved NOA.
SPARE0/SPARE2	Spare. Not used.
SUB-OPR	NOA used to specify subscriber number, operator requested (valid for called party number only).
SUBSCRIBER	NOA used to specify subscriber number.
TEST-LINE	NOA used to specify test line number.
UNKNOWN	NOA is unknown.
VSC	NOA used to specify vertical service code.

Nature of Address Route Profile

The Nature of Address (NOA) Route Profile (noa-route-profile) table is used to support NOA-based routing on the Cisco BTS 10200. This profile defines the NOA route ID. The ID can be assigned to a single Dial Plan Profile table or multiple Dial Plan Profile tables. For token names and description details for the NOA Route Profile table, refer to the Cisco BTS 10200 Softswitch CLI Database.

Nature of Address Route

The Nature of Address (NOA) Route (noa-route) table defines NOA-based routing on the Cisco BTS 10200. When the NOA-based routing is specified in the Dial Plan Profile table, the Cisco BTS 10200 uses the received called NOA to index the NOA Route table and determine the destination ID for further routing. The destination ID can point to a specific route based on the NOA or it can point to a dial plan. When a destination ID points to a dial plan, the received called party number is translated by use of the dial plan. For token names and description details for the NOA Route table, refer to the Cisco BTS 10200 Softswitch CLI Database.

ANI or DNI Service Manipulation

The Cisco BTS 10200 supports DNIS or ANI digit manipulation, it includes digit manipulation or nature of address manipulation or both. When ANI manipulation is defined, it usually refers to the calling party number (CPN) which is normally displayed when a call is terminated to a device with the calling number delivery (CND) feature.

In countries like China, there are additional requirements regarding ANI display. The requirements are to display ANI in the dialable format. The term *dialable* means the user can simply call up the display and hit the dial button to place the call based on the received ANI. This requires ANI to contain an area code or national destination code (NDC) for a long distance call while only displaying subscriber number for a local call. The rules defined above can be used to manipulate ANI digits to the desired format.

When call forwarding has occurred, the ANI manipulation is performed as follows:

- If redirecting party number is available, the redirecting party number is manipulated.
- If redirecting party number is not available, but original called number (OCN) is available, the OCN is manipulated.
- If neither redirecting party number nor OCN is available, the calling party number is manipulated.

The following sections describe the provisioning and actions in each of the three stages in greater detail.

Pretranslation Stage

In the pretranslation stage, the dial-plan-profile table is used to specify if ANI, DNIS, or both are to be manipulated. The purpose of the pre-translation stage is to normalize the digits, as required, during the translation stage.

In addition to manipulation of the ANI/DNIS digits, the pretranslation stage can also be used to determine the NOA of the incoming digits (either ANI or DNIS or both).

The Dial Plan Profile table has been modified to allow provisioning of the ANI and DNIS digit manipulation rules.

Example 1

In the first example, since the first character of the input string matches the specified match-string (^*), the NOA is changed to a vertical service code (VSC).

```
Add digman id=pretrans; rule=1; match-string=^*; replace-string=&; match-noa=any;
replace-noa=vsc;

Enter input string *55#
Match string ^*
Replace string &
MATCHED

Output string=*55#
```

Example 2

In the second example, since the last character of the input string matches the specified match-string (#), the NOA is changed to a VSC.

```
Add digman id=pretrans; rule=2; match-string=#; replace-string=&; match-noa=any;
replace-noa=vsc;

Enter input string *55#
Match string #
Replace string &
MATCHED

Output string=*55#
```

Example 3

In the third example there is no match, so the NOA is not changed and the output-string is the same as the input-string.

```
Add digman id=pretrans; rule=3; match-string=*; replace-string=&; match-noa=any;
replace-noa=vsc;

Enter input string 5555
Match string *
Replace string &
Not MATCHED

Output string=5555
```

Determining NOA for China

The following table can be used to determine the NOA of incoming calls for China.

Table 1-3   NOA Determination for China

Rule No.	Match-String	Replace String	Match-NOA	Replace NOA	Remarks
1	*	&	Any	VSC	If first digit is a *, treat it as a VSC code.
2	#	&	Any	VSC	If first digit is a #, treat it as a VSC code.

Digit Manipulation Profile

The Digit Manipulation Profile (digman-profile) table is used to create unique IDs for digit manipulation. This ID must be created before the Digit Manipulation table can be provisioned. For token names and description details for the Digit Manipulation Profile table, refer to the Cisco BTS 10200 Softswitch CLI Database.

Digit Manipulation

The Digit Manipulation (digman) table is used to perform digit and nature of address (NOA) manipulation. For token names and description details for the Digit Manipulation table, refer to the Cisco BTS 10200 Softswitch CLI Database.

Dial Plan Profile

The Dial Plan Profile (dial-plan-profile) table creates dial-plan-profile ids before they are assigned to subscribers or trunk groups. The dial-plan-profile ID links digit-string entries in the Dial Plan table within a dial plan. Different dial-plan-profile ids are assigned to subscribers and trunk groups. A dial-plan-id must be created in this table before entries can be added to the Dial Plan table. For token names and description details for the Dial Plan Profile table, refer to the Cisco BTS 10200 Softswitch CLI Database.

International Dial Plan Profile

The International Dial Plan Profile (intl-dial-plan-profile) table is used to create unique IDs for international dial plans. This ID must be created before provisioning the International Dial Plan table. For token names and description details for the International Dial Plan Profile table, refer to the Cisco BTS 10200 Softswitch CLI Database.

Translations Stage

The Dial-Plan table or International Dial-Plan table is used during the Translations stage to manipulate only the called party number. The previously existing capability of a simple delete and/or prefix digit manipulation function is supported.

> **Note** Although the simple delete and/or prefix digit manipulation function is still supported, it is no longer necessary, becuase the same function(s) can now be accomplished in the other two stages.

Dial Plan

Dial plans analyze, screen, and route calls based on dialed digits. The Dial Plan (dial-plan) table holds dial plan information for a specific type of call. It defines valid dialing patterns and determines call routing. All records that share a common dial-plan-profile ID are considered a dial plan. For token names and description details for the Dial Plan table, refer to the Cisco BTS 10200 Softswitch CLI Database.

International Dial Plan

The International Dial Plan (intl-dial-plan) table holds international dial plan information for calls to regions outside the North American Numbering Plan (NANP). It contains the country code, minimum and maximum digits, the country name, and the route-grp-id. For token names and description details for the International Dial Plan table, refer to the Cisco BTS 10200 Softswitch CLI Database.

Variable Digit Dialing

Variable digit dialing is used in the Europe where the length of the dialed number can vary from seven digits to ten digits, mainly in Germany. For a given NDC or EC, the DN can vary from the minimum to the maximum specified in the Exchange Code table.

Example:

ndc=349, ec=234; min-digits=7; max-digits=10; (from the Exchange Code table)

DN=3492340 could be a DN

DN=3492341234 could also be a DN belonging to ndc=349, ec=234

DN=349234222 could also be a 9 digit DN

Variable Digit Dialing Provisioning

To provision variable digit dialing according to the example given in the "Variable Digit Dialing" section, take the following steps:

Step 1   add exchange-code ndc=349, ec=234; min-digits=7; max-digits=10;

Step 2   add office-code ndc=349; ec=234; dn-group=0; (For the DN=3492340)

Step 3   add office-code ndc=349; ec=234; dn-group=1xxx; dn-length= 10; (For the 10 digit DNs)

Step 4   add office-code ndc=349; ec=234; dn-group=2xx; (For the 9 digit DNs in the dn group)

Step 5   add office-code ndc=349; ec=234; dn-group=12x; (For the 9 digit DNs)

Routing Stage

The digit manipulations for an outgoing call can be performed using one of the following tables:

- Destination table
- Route table
- Trunk Group table for subscriber termination

The ANI/DNIS digit manipulation specified in the Destination table is applied during the outgoing call setup. The out-pulsing number can also be normalized using the destination table. In addition, if special digit manipulation is required based on the route selected, it can be specified for each trunk group within a route.

If the called number terminates within the Cisco BTS 10200, and the subscriber number points to a trunk group, then digit manipulation rules for ANI, DNIS, or both can be specified in the Trunk Group table.

If the call is an interLATA call or requires Carrier routing (ROUTE-TYPE=CARRIER), the ANI/DNIS digit manipulation IDs specified in the destination table are ignored and carrier-based routing is performed.

> **Note** Although both **pfx_digits** and **del_digits** still work in the Cisco BTS 10200 dial plan table, if the subscriber dials a 7-digit number, but local number portability (LNP) queries require a 10-digit DN, then the digman tables should be set up to add an NPA, making the called number a 10-digit DN before an LNP query is performed.

Destination

The Destination (destination) table defines the call type and the routing information for the dialed digits. Multiple digit strings in the Dial Plan table can use the same destination ID. For token names and description details for the Destination table, refer to the Cisco BTS 10200 Softswitch CLI Database.

For additional information on call types, refer to Appendix A, "Call Types and Subtypes."

Route

The Route (route) table contains a list of up to ten trunk groups for call routing. If all the trunk groups are busy or not available, call processing uses the alt-route-id (if specified) to route the call. The Element Management System (EMS) provisions the Call Agent ID field based on the Trunk Group table. For token names and description details for the Route table, refer to the Cisco BTS 10200 Softswitch CLI Database.

> **Note** The Route table enables the service provider to provision a list of up to 10 trunk groups (TG1 to TG10), and a parameter for selecting the priority of the TGs for routing (TG-SELECTION). The system attempts to route the call on the highest priority TG. If the call cannot be completed on the highest priority TG, the system attempts to use the next (lower priority) TG, a process known as route advance. The system attempts route advance to lower priority TGs up to three times. (Any TG in the list that is administratively out of service is not counted as an attempt.) If all three attempts fail, the call is released, and the system provides a release announcement.

Trunk Group

The Trunk Group (trunk-grp) table identifies the trunk group and maps it to the associated media gateway.

The Cisco BTS 10200 supports the following trunk group types: announcement, CAS, ISDN, SS7 and SOFTSW. The Trunk Group table defines common information based on the trunk group type. The Cisco BTS 10200 supports announcement, CAS, ISDN, SS7 and SOFTSW trunk group profiles. For token names and description details for the Trunk Group table, refer to the Cisco BTS 10200 Softswitch CLI Database.

Policy NXX

The Policy NXX (policy-nxx) table is used when a number services call results in a translated number, carrier ID, translated number and a carrier ID, or a route ID. For token names and description details for the Policy NXX table, refer to the Cisco BTS 10200 Softswitch CLI Database.

> **Note** The original call type is preserved in billing when the routing number is changed by the use of the Policy NXX table.

Call Type Profile

The Call Type Profile (call-type-profile) table defines the call type properties supported by the Cisco BTS 10200. Provision this table if the ALL-CALL-QUERY flag in the LNP-PROFILE table is set to Y and the ACQ-LNP-QUERY token in the Destination table is set to ACQ-BASED-ON-CALL-TYPE. For token names and description details for the Call Type Profile table, refer to the Cisco BTS 10200 Softswitch CLI Database.

For additional information on call types, refer to Appendix A, "Call Types and Subtypes."

Sample Provisioning

Figure 1-3 provides a sample provisioning script for the implementation of the digman functionality.

Figure 1-3   *Sample Provisioning*
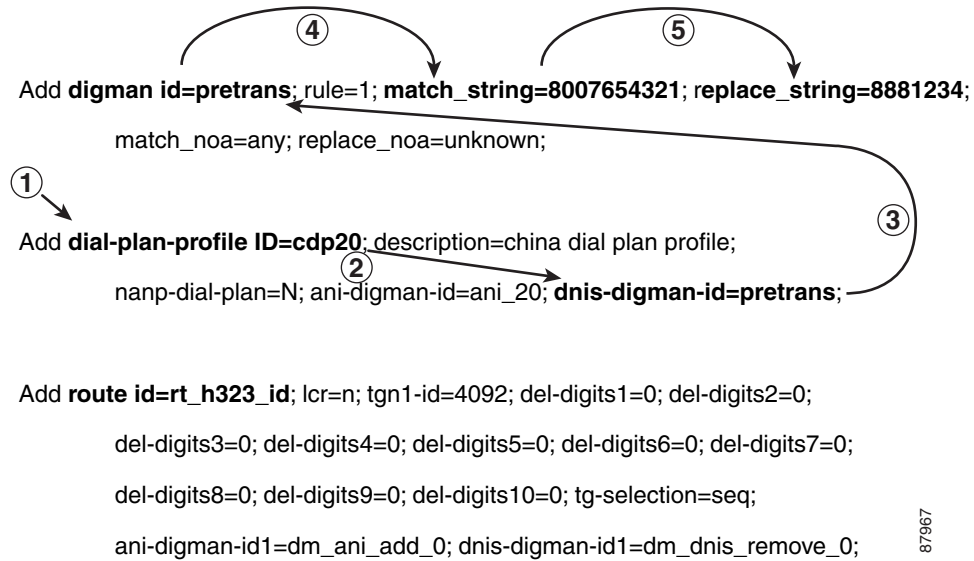
Caller A, using dial-plan-id=cdp20, dials 8007654321. The call hits the dial-plan-profile table (1) and uses the pretranslation entry (2 and 3). It matches on the 800 number (4) and replaces it with 8881234 (5).

Feature Interaction

When a call is terminated to a subscriber with Call Forwarding activated, a feature interaction occurs when calls are forwarded out of the Cisco BTS 10200. In China, the Cisco BTS 10200 uses H.323 signaling to route calls to the PSTN; however, this scenario applies to any call which is forwarded over a non-SS7 signaling interface.

The H.323 protocol and other non-SS7 signaling protocols do not support the Charge Number parameter, Original Called Party Number, or Redirecting Party Number. When call forwarding occurs, the ANI manipulation is performed on the redirecting party number if it is available. The DN of the forwarding party should also be sent as ANI digits and, as such in a forwarding scenario, the called number of the forwarding party should also be manipulated for the call.

Feature Interaction