Dial Plan and Routing

Revised: January 20, 2011, Cisco MGC Software Release 9.5(1)

This chapter provides an overview of the role the dial plan plays in call processing on the Cisco PGW 2200 Softswitch. Dial plans let the Cisco PGW 2200 Softswitch running the MGC software communicate with the Signaling System 7 (SS7) network and with the system components that control media gateways and bearer-traffic routing.

This chapter contains the following sections:

- Dial Plan and Routing Introduction, page 1-1
- Result Analysis, page 1-9
- Pre-analysis, page 1-70
- Number Analysis, page 1-78
- Cause Analysis, page 1-78
- Routing Analysis, page 1-107

The dial plan provisioning processes described in this document apply to all Cisco telephony solutions running the Cisco PGW 2200 Softswitch Release 9.3 and later.

Dial Plan and Routing Introduction

A dial plan lets you manipulate and make decisions based on the incoming call data. The dial plan can perform Pre-analysis, A-number analysis, and B-number analysis for either nailed (signaling) or switched (call control) call routing, and cause analysis. Dial plans and routing are explained in the following sections.

- Dial Plan Overview, page 1-3
- Routing Overview, page 1-6
Figure 1-1 provides a high-level overview of call analysis and routing stages.

**Figure 1-1   MGC Call Analysis and Routing Stages**

Pre-analysis lets you make decisions based on parameters received in the incoming IAM, Setup, or SIP INVITE message and optionally manipulate data within those parameters.

A-number analysis lets you make decisions based on the calling number received in the incoming message and optionally manipulate data based on the calling number. The calling number is the number from which the call is originating. The incoming calling number for A-number analysis might have been previously manipulated within the Pre-analysis stage.

B-number analysis lets you make decisions based on the called number received in the incoming message and optionally manipulate data based on the called number. The called number is the number to which the call is destined. The incoming called number for B-number analysis might have been previously manipulated within the Pre-analysis stage and/or the A-number analysis stage.

In a signaling call environment, Route analysis is not performed because the terminating gateway is already determined based on the incoming trunk.

In a call control environment, decisions are always made in the dial plan about whether the call is switched using the routing analysis stage, or is treated as a signaling call. The routing analysis stage can be initiated from Pre-analysis, B-number analysis, or Cause analysis stages.

Routing analysis lets you direct the call to an outgoing trunk group. Currently the supported entry points into routing are

- Directly into Route analysis
- Conditional route analysis (Release 9.3(2) functionality)
- Percentage based route analysis (Release 9.3(2) functionality)

Cause analysis lets you make decisions based on parameter information received in the release messages, or internally set failure information. These messages can result in a cause message being sent to the gateway, to route advance, and to change to another dial plan with a restart of analysis.
Dial Plan Overview

When creating a dial plan, you must first determine if the call type is signaling or call control. A dialplan can be used in either a signaling or call control configuration and provides different levels of functionality according to the networking environment.

In a signaling environment, the ingress and egress circuits are already fixed at the outset of the call so the dialplan does not finish B-number analysis with a Routing result that provokes Routing analysis, since this is not necessary.

In a call control environment, the call fails unless Pre-analysis or B-number analysis produces a Routing result, since the Routing analysis stage must be invoked to determine egress routes, trunk groups, and trunks provisioned in the Cisco PGW 2200 Softswitch.

For each stage within the dial plan, the resulting actions are selected, based on the incoming parameter values. The incoming parameters are contained in the ISDN User Part (ISUP) IAM, which is included in the Signaling Information Field (SIF) of an SS7 Message Signal Unit (MSU), in the SIP INVITE method, or in the ISDN PRI Setup message.

The actions are referred to as results and can be grouped into result sets consisting of one or more results. Combining different results within result sets provide a flexible mechanism for selecting subsequent analysis actions. This includes the ability to re-enter previous analysis stages. Additionally, result sets can be performed at multiple points of analysis within a stage.

Some common result actions are digit modification, NOA manipulation, and screening. Dial plans support both open and closed numbering plans.

The dial plan can be changed dynamically and the changes take effect with the next call setup.

Dial Plan Selection Overview

The dial plan functionality can handle multiple, independent customer networks, each with its own set of actions. To support this, all dial plans are created with a customer group identifier, called CustGrpId. This customer group identifier is used to associate the first dial plan to a sigpath in a signaling environment or a trunk group in a call control environment.

Figure 1-2 shows how to change from one dial plan to another, based on result sets configured at the various stages of analysis. You can select a new dial plan either by specifying a CustGrpId in a result or by initiating a lookup of a new CustGrpId in a dial plan selection table. See the “Dial Plan Selection” section on page 1-82 for more information. Multiple dial plan functionality allows up to 10 subsequent dial plans to be used during analysis of a single call.
A Cisco PGW 2200 Softswitch can be presented with calls from public switched telephone network (PSTN) service providers that could be handled in different ways. Here are some examples:

- National calls tandem switched through the Cisco PGW 2200 Softswitch
- International calls requiring different treatment before being tandem switched by the Cisco PGW 2200 Softswitch
- Re-seller calls tandem switched through the Cisco PGW 2200 Softswitch
- Private branch exchange (PBX) calls requiring breakout to the PSTN
- Internet calls terminating over ISDN primary rate interfaces (PRIs) hosted on the Cisco PGW 2200 Softswitch

Calls originating from a virtual private network (VPN) on a PBX (PRI) ingress can be routed within a “local” dial plan (the dial plan for the VPN) by analyzing extension digits; or calls can be routed out over the PSTN with a full national number by dialing a PSTN access code, such as 9.

For reseller type calls the customer line is “virtual” to the re-seller service provider and is known only to that provider by the calling party number (A-number). Thus, the required switching actions must be determined according to the A-number; hence the requirement to change dial plans according to this number. It must be understood that in such a scenario the volume of A-numbers is constrained by the level of usage of the Cisco PGW 2200 Softswitch, as described later.

The system must support the ability to start call processing within the dial plan defined against an ingress trunk group or sigpath and then, depending on Pre-analysis, A-number analysis, B-number analysis, or Cause analysis identify a new dial plan to continue call processing.
Pre-analysis Overview

The Pre-analysis stages, shown in Figure 1-3, are as follows:

- A/B-number NOA and NPI Analysis (NOA/NPI) for calling number (A-number)
- Calling Party Category Analysis (CPC)*
- Transmission Medium Requirement Analysis (TMR)*
- A/B-number NOA and NPI Analysis (NOA/NPI) for called number (B-number)
- Transit Network Selection Analysis (TNS)*
- NANP B-Number Normalization

* Indicates MGC software Release 9.3(2) functionality.

Figure 1-3 Pre-analysis Stages
The initial analysis request, made after the reception of an SS7 IAM, ISDN PRI setup, or SIP INVITE message, is called Pre-analysis. Pre-analysis, if required, is performed according to the data in the received message. Pre-analysis enables you to perform calling party category (CPC) analysis, transmission medium requirement (TMR) analysis, NOA/NPI analysis, transit network selection (TNS), and North American Numbering Plan (NANP) number normalization before number analysis.

Each Pre-analysis stage is completed and leads to the next Pre-analysis stage unless there is analysis failure or a blacklist result. Once all Pre-analysis stages are completed, the result handling is completed including any dial plan changeover before the call goes to the next analysis stage.

**Number Analysis Overview**

Number analysis is performed following the completion of Pre-analysis. Number analysis analyzes each digit in the A-number (calling number), optionally the Redirecting number, and finally the B-number (called number) to determine if any action should be taken.

**Cause Analysis Overview**

Cause analysis is performed when a release (REL) message is received, or when a failure of some kind has occurred implying that the call must be released. The cause code value or the combined cause code and location code values are analyzed to provide a cause code that provokes rerouting of the call to another switch by the preceding switch, rerouting of the call to an announcement server, reattempt and redirecting, or call release.

**Routing Overview**

The objective of a dial plan, in a call control scenario, is to establish a connection or circuit between the calling number (A-number) and the called number (B-number). Here are definitions for four important call routing terms:

- **Trunk**—A trunk (or circuit), in Cisco PGW 2200 Softswitch terms, is a single TDM voice channel (DS0). It is a physical connection between two points through which a call can be established.

- **Trunk group**—A trunk group is a collection of trunks (or circuits). For the sake of simplicity, Cisco PGW 2200 Softswitch trunk groups are often arranged exactly the same as the trunk groups on the switches on the opposing ends of the packet network.

- **Route**—The route defines the path that a call uses. It might be a collection of trunk groups with the same destination, or a logical path over a packet network fabric.

- **Route List**—A route list is a collection of routing alternatives that can be used to transport a call between the origination and the destination points. Individual routes within a route list can connect the same two origination and destination points, but over different physical paths.
SS7 Call Routing

The dial plan is the primary determinant of how a call is routed from its origination to its termination through a Cisco PGW 2200 Softswitch-controlled packet-switched network. Figure 1-4 is a simplified illustration of the sequence of events that occur in routing a call from its origination to its termination.

**Figure 1-4  MGC Call Routing Sequence**

The Cisco PGW 2200 Softswitch routing functionality includes the following, as shown in Figure 1-4:

- When the MGC is used for tandem (transit) applications, all calls originate or terminate outside the MGC-controlled packet network.
- The MGC receives and analyzes signaling messages, either SS7 or ISDN PRI, determines ingress and egress gateways, and selects the egress trunks (or circuits) to external TDM switches and networks.
- The MGC controls the ingress and egress media gateways on the packet network edges; however, it does not control the route taken within the packet network.
- The MGC connects the ingress trunk to the egress trunk and routes the call from the origination to the destination.

The Cisco PGW 2200 Softswitch supports random distribution of calls across multiple trunk groups belonging to a particular route. Enabling or disabling random distribution is supported on a route-by-route basis.

Call routing can be accomplished based on factors such as, including the NOA value in the incoming IAM or Setup message, or combining the NOA value and the incoming NPI value. If routing is not determined solely based on Pre-analysis, then number analysis is performed.

Once a route is chosen, the Cisco PGW 2200 Softswitch selects a trunk group and an available trunk (circuit). If no trunk is available, the Cisco PGW 2200 Softswitch releases the call with a cause code indicating all circuits are busy. As shown in Figure 1-1, calls can also be rejected at any point during analysis and released with an appropriate cause code, or routed to an announcement server that informs the caller why the call was not completed.
Call Routing to an IP Endpoint

Figure 1-5 is a simplified illustration of the sequence of events that occur in routing a call to an IP endpoint.

**Figure 1-5 MGC Call Routing Sequence to an IP Endpoint**

The Cisco PGW 2200 Softswitch routing functionality includes the following, as shown in Figure 1-5:

- When the MGC is used to terminate calls to a SIP or H.323 endpoint; calls terminate inside the packet network.
- The MGC receives and analyzes signaling messages, either SS7 or ISDN PRI, determines the ingress gateway, and selects the egress SIP or H.323 signaling path.
- The MGC controls the ingress media gateways on the ingress packet network edge and signals to the SIP or H.323 endpoint; however, it does not control the route taken within the packet network.
- The MGC connects the ingress trunk to the egress SIP or H.323 endpoint and routes the call from the origination to the destination.
Call Routing from an IP Endpoint

Figure 1-6 is a simplified illustration of the sequence of events that occur in routing a call from an IP endpoint.

The Cisco PGW 2200 Softswitch routing functionality includes the following, as shown in Figure 1-6:

- When the MGC is used to originate calls from a SIP or H.323 endpoint; calls originate inside the packet network.
- The MGC receives and analyzes signaling messages, either SIP or H.323, determines the egress gateway and selects the egress trunks (or circuits) to external TDM switches and networks.
- The MGC controls the egress media gateway on the packet network edge; however, it does not control the route taken within the packet network.

These are basic call processing and routing functions of a dial plan. Creating a complete, efficient, and comprehensive dial plan requires thorough planning and foresight. Organization can simplify dial plan implementation.

Result Analysis

Result analysis lets you group actions into result sets that can be attached at different points of analysis. The main attachment points are Pre-analysis, A-number analysis, B-number analysis, and Cause analysis.

When configuring results, there are certain result types, which require extra configuration to provide additional data, that enables the required action. The following are examples of two such result types.
• Number modification where digits are to be inserted into a number. These new digits must be configured first and stored in data before the actual result, which will make use of these digits, is defined. For example, if the B-number is 4841234 and the intention with a B-number modification (BMODDIG result) is to insert 703 at the front of the number, the 703 digit string must be created first. Once the digit string is created, the actual B-number modification result can be defined by means of the 703 digit string data. This is more fully described in the following section.

• When A-number screening is required, if this action is triggered from the B-number digit analysis, it is necessary to identify the database area that contains the A-number screening data for calls destined to this particular B-number. The database area is called the service name. The service name data must be defined separately before the actual A-number screening result is defined. Once again the two following sections explain this more fully.

Finally, when you configure results to invoke Routing actions, there are three types of Routing results ROUTE, COND_ROUTE*, and PERC_ROUTE* which are more fully explained in the following sections.

*Indicates software Release 9.3(2) functionality.

Digit Modification String

The digit modification string entry, Example 1-1, defines the digit modification string for a digit modification name. The digit modification string inserts the specified number of digits into the calling number (A-number) or called number (B-number) at the application point specified in the AMODDIG or BMODDIG result type. Table C-4 in Appendix C, “Dial Planning Worksheets,” can be copied and filled in to document the digit modification names and digit modification strings used in your dial plan.

You can set up the digit modification with a CustGrpID of t001, a digit modification name of digmod3, and a digstring value of 703486.

Example 1-1  Digit Modification String Example

<table>
<thead>
<tr>
<th>Digit Modification Name</th>
<th>Digit Modification String</th>
</tr>
</thead>
<tbody>
<tr>
<td>digmod1</td>
<td>703484</td>
</tr>
<tr>
<td>digmod2</td>
<td>703485</td>
</tr>
<tr>
<td>digmod3</td>
<td>703486</td>
</tr>
</tbody>
</table>

Note  Digit modification names are limited to 20 alphanumeric characters. Spaces are not allowed.

Service Name

Example 1-2 gives service name examples. Table C-5 in Appendix C, “Dial Planning Worksheets,” can be used to plan the service name.

A service name, shown in Example 1-2, provides additional call screening capabilities. Thus, calls made from a B-number may be allowed to dial Washington and FreePhone, but not allowed to dial TollLine.
Example 1-2  Service Name Example

<table>
<thead>
<tr>
<th>Service Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Washington</td>
</tr>
<tr>
<td>FreePhone</td>
</tr>
<tr>
<td>TollLine</td>
</tr>
</tbody>
</table>

**Note**  Service names are limited to 10 alphanumeric characters. Spaces are not allowed in service names.

## Results

A result is a specific action on the Cisco PGW 2200 Softswitch. When you configure a result, you set the result type for this result. You also set values for data words in this result.

Table 1-1 lists all the result type names and their data words. Result types prescribe the actions that are taken when the last analyzed digit in a digit string is reached. See the “Result Type Definitions” section on page 1-17 section following this table for definitions of result types and their associated data words.

**Note**  The result number is only seen in an MDL trace. The result number is not provisionable.

### Table 1-1  Result Type Definitions

<table>
<thead>
<tr>
<th>Result Number</th>
<th>Result Type</th>
<th>Data Word 1</th>
<th>Data Word 2</th>
<th>Data Word 3</th>
<th>Data Word 4</th>
<th>Analysis Points</th>
<th>Result Type Valid For</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>DIGIT_REQ</td>
<td>Num. of digits</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>2</td>
<td>ROUTE</td>
<td>RouteListName</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X X X X</td>
</tr>
<tr>
<td>3</td>
<td>INC_NUMBERING</td>
<td>Numbering Type</td>
<td>Min. digits</td>
<td>Max. digits</td>
<td>0 (not used)</td>
<td>X</td>
<td>X X</td>
</tr>
<tr>
<td>4</td>
<td>BMODDIG</td>
<td>Application Point</td>
<td>Num. of digits to remove</td>
<td>Modification Name</td>
<td>0 (not used)</td>
<td>X</td>
<td>X X X X</td>
</tr>
<tr>
<td>5</td>
<td>AMODDIG</td>
<td>Application Point</td>
<td>Num. of digits to remove</td>
<td>Modification Name</td>
<td>Conditional Indicator</td>
<td>X</td>
<td>X X</td>
</tr>
<tr>
<td>6</td>
<td>CAUSE</td>
<td>Cause Code</td>
<td>Location value</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X X X X</td>
</tr>
<tr>
<td>7</td>
<td>FACILITY</td>
<td>type</td>
<td>treatment</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X X</td>
</tr>
<tr>
<td>8</td>
<td>ANNOUNCEMENT</td>
<td>Announcement ID</td>
<td>Local/Remote</td>
<td>RouteListId</td>
<td>Announceme nt Data</td>
<td>X</td>
<td>X X X X</td>
</tr>
<tr>
<td>Result Number</td>
<td>Result Type</td>
<td>Data Word 1</td>
<td>Data Word 2</td>
<td>Data Word 3</td>
<td>Data Word 4</td>
<td>Analysis Points</td>
<td>Result Type Valid For</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------------</td>
<td>-------------------------------------------</td>
<td>------------------------</td>
<td>--------------------------------------</td>
<td>---------------------------------</td>
<td>-----------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>10</td>
<td>CHARGE</td>
<td>TariffRate/ Dest for Charging/ Charge Band Number/ Charge Unit</td>
<td>Scale Factor</td>
<td>ChargeData Discriminator</td>
<td>Charge Type</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>11</td>
<td>CPC_REQ</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>12</td>
<td>CLI_REQ</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>13</td>
<td>BSM_REQ</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>14</td>
<td>FSM_REQ</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>15</td>
<td>A_NUMBER_TYPE</td>
<td>A-number Type</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X X X</td>
</tr>
<tr>
<td>16</td>
<td>B_NUMBER_TYPE</td>
<td>B-number Type</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X X X X X</td>
</tr>
<tr>
<td>17</td>
<td>OTG_NUMBERING</td>
<td>Numbering Type</td>
<td>Min. digits</td>
<td>Max. digits</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>18</td>
<td>BLACKLIST</td>
<td>Screening Criteria</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X X X X</td>
</tr>
<tr>
<td>19</td>
<td>CLI_NBR_LENGTH</td>
<td>Numbering Type</td>
<td>Min. digits</td>
<td>Max. digits</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>21</td>
<td>ROUTE_PREFERENCE</td>
<td>Route Pref</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>22</td>
<td>IN_TRIGGER</td>
<td>Service Type</td>
<td>SCP/STP Index</td>
<td>Min digits Req</td>
<td>Timer</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>23</td>
<td>SCREENING</td>
<td>Screen Type</td>
<td>Service Name</td>
<td>Pass_DpIdx</td>
<td>Fail_DpIdx</td>
<td>X</td>
<td>X X</td>
</tr>
<tr>
<td>24</td>
<td>DATA_EXCHANGE</td>
<td>Action Type</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>25</td>
<td>E_PORTED_NUM</td>
<td>Number of digits to remove Use partial number</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>E_ROUTE_NUM</td>
<td>Number of digits to remove</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>27</td>
<td>TERM_INFO</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>28</td>
<td>TESTCALLDETECTED</td>
<td>TestLineType</td>
<td>TestLine Duration</td>
<td>TestLineName</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>31</td>
<td>ADDRESSCLASS</td>
<td>Geographic</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>32</td>
<td>WHITELIST</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>33</td>
<td>NEW_DIALPLAN</td>
<td>CustGrpId</td>
<td>Analysis Type</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X X X X</td>
</tr>
<tr>
<td>34</td>
<td>A_NUM_DP_TABLE</td>
<td>searchMin</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>35</td>
<td>RTRN_START_ANAL</td>
<td>Number of digits to remove</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X X</td>
</tr>
</tbody>
</table>
### Table 1-1  Result Type Definitions (continued)

<table>
<thead>
<tr>
<th>Result Number</th>
<th>Result Type</th>
<th>Data Word 1</th>
<th>Data Word 2</th>
<th>Data Word 3</th>
<th>Data Word 4</th>
<th>Analysis Points</th>
<th>Result Type Valid For</th>
</tr>
</thead>
<tbody>
<tr>
<td>36</td>
<td>CHARGEORIGIN</td>
<td>Charge Origin</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>37</td>
<td>CG_PRES_IND</td>
<td>Presentation Indicator</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X X</td>
</tr>
<tr>
<td>38</td>
<td>CALL_CUTOFF_TIME</td>
<td>callcutoffvalue</td>
<td>values: 1-48, 1-2880, or 1-172800</td>
<td>callcutoffunits</td>
<td>values: 0, 1, or 2</td>
<td>0 (not used)</td>
<td>X X X</td>
</tr>
<tr>
<td>42</td>
<td>RETRY_ACTION</td>
<td>RetryType (1 to 3)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>43</td>
<td>COND_ROUTE</td>
<td>CondRteName</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X X X X</td>
</tr>
<tr>
<td>44</td>
<td>MGCPDIALPKG</td>
<td>Digital, Analog, or Dynamic</td>
<td>0 or 1</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X X</td>
</tr>
<tr>
<td>45</td>
<td>CPCMOD</td>
<td>Integer (0–255) Calling party CPC parameter</td>
<td>Integer 0 Default-calling party CPC</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X X</td>
</tr>
<tr>
<td>45</td>
<td>CPCMOD</td>
<td>Integer (0-3) Called party CPC parameter</td>
<td>Integer 1 Called party CPC</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X X</td>
</tr>
<tr>
<td>46</td>
<td>CC_DIG</td>
<td>CCModName</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>47</td>
<td>CODEC</td>
<td>CodecStringName</td>
<td>Action</td>
<td>CodecStringType</td>
<td>0 (not used)</td>
<td>X</td>
<td>X X X X</td>
</tr>
<tr>
<td>48</td>
<td>PERC_ROUTE</td>
<td>PercRteName</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X X X</td>
</tr>
<tr>
<td>49</td>
<td>PNMODDIG</td>
<td>Application point</td>
<td>Number of digits to remove</td>
<td>Modification Name</td>
<td>0 (not used)</td>
<td>X</td>
<td>X X</td>
</tr>
<tr>
<td>50</td>
<td>PN_NUMBER_TYPE</td>
<td>Internal NOA value (0-53)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X X</td>
</tr>
<tr>
<td>51</td>
<td>PN_PRES_IND</td>
<td>1 = Restricted 2 = Allowed 3 = Unavailable</td>
<td>Local/Remote RouteListId</td>
<td>AnnData</td>
<td>X</td>
<td>X X</td>
<td></td>
</tr>
<tr>
<td>52</td>
<td>CG_SCREEN_IND</td>
<td>1 = Network Provided 2 = UPVP 3 = UPNV 4 = UPVF 5 = spare</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X X</td>
</tr>
</tbody>
</table>
### Table 1-1  Result Type Definitions (continued)

<table>
<thead>
<tr>
<th>Result Number</th>
<th>Result Type</th>
<th>Data Word 1</th>
<th>Data Word 2</th>
<th>Data Word 3</th>
<th>Data Word 4</th>
<th>Analysis Points</th>
<th>Result Type Valid For</th>
</tr>
</thead>
</table>
| 53            | PN_SCREEN_IND       | 1 = Network Provided  
2 = UPVP  
3 = UPNV  
4 = UPVF  
5 = spare1 | 0 (not used) | 0 (not used) | 0 (not used) | X     | X     | X             |
| 54            | A_NUM_NPI_TYPE      | Internal NPI value (1–10) | 0 (not used) | 0 (not used) | 0 (not used) | X     | X     | X             |
| 55            | CG_PN_COPY          | Index to Network Numbering string | 0 (not used) | 0 (not used) | 0 (not used) | X     | X     | X             |
| 56            | PN_NPI_TYPE         | Internal NPI value (0–10) | 0 (not used) | 0 (not used) | 0 (not used) | X     | X     | X             |
| 57            | RMODDIG             | Application point  
Number of digits to remove  
Modification Index  
Remove leading digits | 0 (not used) | 0 (not used) | 0 (not used) | X     | X     | X             |
| 58            | R_NUMBER_TYPE       | Remote Number Type  
0 = OCN NOA is not updated based on redirecting number.  
1 = OCN NOA is updated based on redirecting number. | 0 (not used) | 0 (not used) | 0 (not used) | X     | X     | X             |
| 59            | ATM_ORIG_PROFILE    | AtmProfIdx  
Action | 0 (not used) | 0 (not used) | 0 (not used) | X     | X     | X             |
| 60            | ATM_TERM_PROFILE    | AtmProfIdx  
Action | 0 (not used) | 0 (not used) | 0 (not used) | X     | X     | X             |
| 61            | SCRIPT              | ScriptId  
CallType  
AcmReqdInd. N/A | 0 (not used) | 0 (not used) | 0 (not used) | X     | X     | X             |
| 62            | CHARGE_MODE_IND     | ChargeModeInd | 0 (not used) | 0 (not used) | 0 (not used) | X     | X     | X             |
| 63            | CHARGE_IND          | ChargeInd | 0 (not used) | 0 (not used) | 0 (not used) | X     | X     | X             |
| 64            | B_NBR_MOD_MWI       | MWI ON  
MWI OFF | 0 (not used) | 0 (not used) | 0 (not used) | X     | X     | X             |
| 65            | IN_SERVICE_KEY      | InServiceKey  
Global Title Digits Type  
Digits Name | 0 (not used) | 0 (not used) | 0 (not used) | X     | X     | X             |
| 66            | LOC_LABEL           | Location Label | 0 (not used) | 0 (not used) | 0 (not used) | X     | X     | X             |
| 67            | OVERRIDE_CALLIM     | 0 (not used) | 0 (not used) | 0 (not used) | 0 (not used) | X     | X     | X             |
### Table 1-1  
**Result Type Definitions (continued)**

<table>
<thead>
<tr>
<th>Result Number</th>
<th>Result Type</th>
<th>Data Word 1</th>
<th>Data Word 2</th>
<th>Data Word 3</th>
<th>Data Word 4</th>
<th>Analysis Points</th>
<th>Result Type Valid For</th>
</tr>
</thead>
<tbody>
<tr>
<td>69</td>
<td><strong>NUM_TRANS</strong></td>
<td>Service Key</td>
<td>Number Type</td>
<td>NOA</td>
<td>Dial Plan</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>70</td>
<td><strong>E911PROF</strong></td>
<td>Route Pref</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>71</td>
<td><strong>ORIG_VPN_ID</strong></td>
<td>VPN ID</td>
<td>On-net index</td>
<td>Off-net index</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>72</td>
<td><strong>DTMFCAP</strong></td>
<td>DTMF Capability</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>73</td>
<td><strong>BCMOD</strong></td>
<td>BC name</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>74</td>
<td><strong>HLCMOD</strong></td>
<td>HLC name</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>75</td>
<td><strong>DB_XLATED</strong></td>
<td>searchMin</td>
<td>matchNewDp</td>
<td>nonMatchedNewDp</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>76</td>
<td><strong>REDIRECT</strong></td>
<td>serviceKey</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>77</td>
<td><strong>IP_SOURCE_SCREEN</strong></td>
<td>screenType</td>
<td>serviceName</td>
<td>foundSetName</td>
<td>notFoundSetName</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>78</td>
<td><strong>IP_DEST_TRANS</strong></td>
<td>inputAndAction</td>
<td>serviceName</td>
<td>foundSetName</td>
<td>notFoundSetName</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>79</td>
<td><strong>IP_SET_SOURCE_DM</strong></td>
<td>dmnString</td>
<td>applicationStatus</td>
<td>applyTo</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>80</td>
<td><strong>IP_ROUTE_SEL</strong></td>
<td>inputDataType</td>
<td>serviceName</td>
<td>foundSetName</td>
<td>notFoundSetName</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>81</td>
<td><strong>DRP_EXIT</strong></td>
<td>drpExitType</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>82</td>
<td><strong>SIPTNS</strong></td>
<td>Circuit Code Value</td>
<td>(0-15)</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>83</td>
<td><strong>SIPI_CONTROL</strong></td>
<td>Enable the route</td>
<td>preference</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>84</td>
<td><strong>GATEWAYPOOL</strong></td>
<td>Ingress gateway</td>
<td>pool ID</td>
<td>AnchorMedia property</td>
<td>value on the ingress side</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>85</td>
<td><strong>VIDEO_ALLOWED</strong></td>
<td>Allows or prohibits</td>
<td>video calls</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>86</td>
<td><strong>DEFAULT_TMR</strong></td>
<td>Specifies the</td>
<td>default TMR value</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>87</td>
<td><strong>CALL_TAG</strong></td>
<td>Call tag list</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>88</td>
<td><strong>RESULT</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Table 1-1 Result Type Definitions (continued)

<table>
<thead>
<tr>
<th>Result Number</th>
<th>Result Type</th>
<th>Data Word 1</th>
<th>Data Word 2</th>
<th>Data Word 3</th>
<th>Data Word 4</th>
<th>Analysis Points</th>
<th>Result Type Valid For</th>
</tr>
</thead>
<tbody>
<tr>
<td>89</td>
<td>CALL_REPORT</td>
<td>Call severity:</td>
<td>Predefined text that you configure to be sent as part of the SNMP</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>90</td>
<td>PREFIX_CONVERT</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>0 (not used)</td>
<td>X</td>
<td>X</td>
</tr>
</tbody>
</table>

Note: All result matches for a digit string are added together and only duplicate result types are overwritten by the longest match.

Operation of Intermediate Result Types

Most of the result types listed in Table 1-1 are classified as “intermediate” result types. Intermediate result types are inserted in result sets; however, they do not signify the end of the analysis. They work throughout the analysis and there is a possibility that other intermediate result types might be encountered further on in the analysis, which can result in the overwriting of a previous result or value.

Intermediate result types provide the ability to provision multiple occurrences of the same result as you go further into the analysis. With intermediate result types the analysis module retrieves them and flags their presence ready for processing. If another intermediate result of the same type is retrieved later in the processing, the new data overwrites the previous data and the last retrieved result becomes the one that is applied.

Note: Intermediate result types can be followed with another intermediate result type or with an end point result type. When an end point result type is encountered in a result set the analysis is complete. An end point result type cannot be followed by any other result type and no more results or result sets can be connected further on in the analysis. End point result types currently used include CAUSE, ANNOUNCEMENT, BLACKLIST, WHITELIST, and MGCPDIALPKG.

For example, intermediate result types can allow you to provision a route to an operator center based on the digit string 703 in the called number (B-number). Later in the analysis you can provision more precise routings for calls that include the 703 digit string, such as a ROUTE result for longer digit strings such as 703484, which routes the call to route 1, 703544, which routes the call to route 2, and so on. The longest string matched wins; however, if you don’t get a longer match, then the earlier route based on the shorter 703 digit string is taken.

Depending on the analysis area that invokes them, the AMODDIG and BMODDIG result types have different functions.
Result Type Definitions

The following paragraphs contain definitions of the result types listed in Table 1-1.

ADDRESSCLASS

The ADDRESSCLASS result type is returned from B-number analysis (the called number) indicating whether the number is geographic or nongeographic. This result type can be encountered during B-number analysis and indicates the class of the B-number. The ADDRESSCLASS result type value indicates the class of address.

Valid ADDRESSCLASS values are
- 0 = Geographic (default)
- 1 = Non-geographic

It is possible to encounter more than one ADDRESSCLASS result for a given B-number and all these results are applied to the B-number. This allows for the addition of future new ADDRESSCLASS results that might not be mutually exclusive.

AMODDIG

The AMODDIG result type is for digit modification on the A-number. You can remove a specified number of digits from any point in the A-digit string and replace them with whatever digits are required. Here is an example of the A-number modification:

If you get result type AMODDIG to modify the A-number, you receive the following datawords:
- Application point—The point (digit) in the digit string to begin applying the modification. The range is from 1 through the total number of digits in the digit string (32 maximum). Entering a value of “98” causes the removal of digits to begin at the end of the digit string and move backward.
- Number of digits to remove—The range is from 0 through the number of digits remaining in the digit string from the application point (32 maximum). To remove the entire number, regardless of the number of digits it contains, enter the value “99” for this dataword.
- Modification name—If required, this is a name that specifies the digit modification string that is to be inserted beginning at the application point.
- ConditionalInd—Provides an indication of conditional modification. 0 indicates unconditional modification and 1 indicates presentation restriction dependent. (Added in software Release 9.5(2).)

Dataword rules:
- Dataword1 must be 1 through 32 or 98.
- Dataword2 must be 0 through 32 or 99.
- Dataword3 must be 0 or an existing digit modification name.
- If dataword4 is 1, then dataword 2 is not allowed and should be 0.
- If dataword 4 is 0, then allow dataword 2 as normal.

For example, if the application point = 1, the number of digits to remove = 5, and the modification name gives a result of 1321, then begin at the start of the digit string, remove 5 digits, and replace them with the digit string 1321. This yields the two following A-number values:
- A-number received pre-analysis = 01444 567891
- A-number post analysis = 1321 567891
Here is another example. If the application point = 98, the number of digits to remove = 4, and the modification name gives a result of 1321, then begin at the end of the digit string, remove 4 digits, and replace them with the digit string 1321. This yields the two following A-number values:

- A-number received pre-analysis = 1234567891
- A-number post analysis = 1234561321

Depending on the analysis area that invokes it, the AMODDIG result type has different functionality. The following are examples of this different functionality.

- In Pre-Analysis there are currently four serial stages that can produce the AMODDIG result type. In Pre-analysis, the results are cumulative. For example, if the CPC stage generates an AMODDIG result type, then the A-number is modified according to the result and this modified number then is the new A-number passed as input to the next Pre-analysis stage (TMR analysis). If the TMR analysis provokes another AMODDIG result type, then it further modifies the number and so on. Even though multiple modifications like this would seem excessive and unnecessary, the capability exists to ensure the required flexibility is provided.

- In Number analysis (A-number or B-number), functionality is different. Here digit analysis is applied (digit by digit), and it is possible to have the AMODDIG result type at multiple points if required. However, only the last modification result type is applied.

**Note**
Digit modification is applied to the initial number input to this analysis stage. There is no cumulative digit modification performed.

For example, if the received A-number is 1234 and at “1” an AMODDIG result type is received making the number 441234, the digit string is modified and analysis continues according to the digit analysis configuration. If another AMODDIG result type is received at 1234, making the number 551234, the earlier AMODDIG result type (“1”) is discarded and the number now sent forward is 551234.

**ANNOUNCEMENT**

The ANNOUNCEMENT result type provides an announcement ID, local or remote indication, and a route ID. These fields are defined as follows:

- **AnnouncementId**—Indicates the identity value corresponding to the announcement identity (or tone identity) that is played to the caller. This is one of the two access keys for which the table is searched. It is a 4-digit integer value.
- **Local/Remote**—Indicates if the Announcement is to be played on a local gateway or routed to a remote announcement server elsewhere in the network. Values: 0—Local (gateway), 1—Remote (gateway).
- **RouteListId**—Indicates the RouteListId that is used to route to a remote announcement server. This dataword is applicable only when dataword2 is set to remote (1).
- **AnnData**—Enables the switching off of a trunk group property announcement for certain A-numbers or B-numbers. It also enables the applying of an announcement for certain A-numbers or B-numbers if no trunk group property has been configured. Values are 0-Off, 1-Interim announcement on, or 2-Final announcement on. This dataword is applicable only when dataword2 is set to local (0).
A_NUM_DP_TABLE
The A_NUM_DP_TABLE result type is relevant to Pre-analysis, A-number, and B-number analysis. Dial plan selection can be triggered by Pre-analysis, the A-number, or the called party number (B-number). The Cisco PGW 2200 Softswitch searches the dial plan for a match on the A-number and, if found, a new dial plan identity is returned that is then used to continue call processing. An external tool encapsulating ttBulkCp supports fast importing/exporting of ported numbers.

This result type has the searchMin dataword. The searchMin dataword indicates how far to search back in the number when longest matching.

A_NUMBER_TYPE
The A_NUMBER_TYPE result type lets you change the A-number type NOA from that presented in the IAM or Setup message. The value given as data in the result type (dataword1) is the Cisco PGW 2200 Softswitch internal call context value for the NOA relating to the A-number. This result type is available to A-number analysis.

Note
The NOA value needs to be the MGC internal value and not the protocol-specific value. See Appendix A, “NOA and NPI Codes, CPC and TMR Values” for specific protocol values.

A_NUM_NPI_TYPE
The A_NUM_NPI_TYPE result type is for CgPN; PN (GN-ACgPN) should be mapped from the original CgPN if it was populated by a swap; or, if it is a new provision, use a default value (E.164).

Dataword1 indicates the internal NPI value. The value range is 1 through 10. The valid values are:

- 1 = NPI_NONE
- 2 = NPI_E164
- 3 = NPI_DATA
- 4 = NPI_TELEX
- 5 = NPI_PNP
- 6 = NPI_NATIONAL
- 7 = NPI_TELEPHONY
- 8 = NPI_MARITIME_MOBILE
- 9 = NPI_LAND_MOBILE
- 10 = NPI_ISDN_MOBILE

ATM_ORIG_PROFILE
The ATM_ORIG_PROFILE result type is used to deliver a profile list configured according to the Service Level Agreement requirements for the originating side. The ATM_ORIG_PROFILE result type has the following datawords:

- Dataword1—AtmProfIdx provides an index value that is used to read the ATM Profiles table from the routeAnalysis.dat file. This enables retrieving a list of ATM profiles for use in the profile negotiation process.
- Dataword2—Action provides an indication as to whether this profile list is to be considered preferred or mandatory. Values are 0 (mandatory) or 1 (preferred).

Possible profile entries are
- ITU1
ATM_ORIG_PROFILE provisioning for analysis results can be performed for either the A-numbers or the B-numbers.

ATM_TERM_PROFILE
The ATM_TERM_PROFILE result type is used to deliver a profile list configured according to the Service Level Agreement requirements for the terminating side. The ATM_TERM_PROFILE result type has the following datawords:

- Dataword1—AtmProfIdx provides an index value that is used to read the ATM Profiles table from the routeAnalysis.dat file. This enables retrieving a list of ATM profiles for use in the profile negotiation process.
- Dataword2—Action provides an indication as to whether this profile list is to be considered preferred or mandatory. Values are 0 (mandatory) or 1 (preferred).

Possible profile entries are
- ITU1
- ITU2
- ITU3
- ITU7
- ITU8
- ITU12
- Custom100
- Custom101
- Custom110
- Custom200

ATM_TERM_PROFILE provisioning for analysis results can be performed for either the A-numbers or the B-numbers.

BCMOD
The BCMOD result type allows you to modify the Bearer Capability of outgoing Initial Address Message (IAMs) based on the dialed Called Party Number. You can provision this result type using A- and B-number analysis.

The BCMOD result type has the BC name dataword. The BC name dataword indicates the Bearer Capability name, such as “fax-bc01.”
BLACKLIST

The BLACKLIST result type provides the basic ability to terminate a call during Pre-analysis and number analysis. If this result is received, the call is immediately released with the cause value IC_BLACKLIST_CL1_MATCHED (which may be changed by the protocol when the Cisco PGW 2200 Softswitch sends the release message to the line). The call is terminated immediately, so there is no screening involved with this result type.

The possible result types (screening criteria) and their application are as follows:

- 1 = Calling Line Identity (CLI)—Analysis of the A-number reveals that this calling line is restricted. It is only supported in A-digit analysis.
- 2 = Dialed Address—Analysis of the B-number reveals that this called line is restricted. It is supported only in B-digit analysis.
- 3 = Calling Party Category (CPC)—Analysis of only the CPC stage of Pre-analysis.
- 4 = Nature of Address (NOA)—Pre-analysis, A-number, and B-number analysis reveal that this calling line is restricted due to its NOA value.
- 5 = Transmission medium requirement (TMR)—Analysis of only the TMR stage of Pre-analysis.
- 6 = Transit network selection (TNS)—Analysis of only the TNS stage of Pre-analysis.

BMODDIG

The BMODDIG result type is for digit modification on the B-number. You can remove a specified number of digits from any point in the B-digit string and replace them with whatever digits are required. Here is an example of the B-number modification:

If we get result type BMODDIG to modify the B-number, we receive the following datawords:

- **Application point**—The point (digit) in the digit string that the Cisco PGW 2200 Softswitch begins applying the modification. The range is from 1 through the total number of digits in the digit string (32 maximum). Entering a value of “98” causes the removal of digits to begin at the end of the digit string and move backward.

- **Number of digits to remove**—The range is from 0 through the number of digits remaining in the digit string from the application point (32 maximum). To remove the entire number, regardless of the number of digits it contains, enter the value “99” for this dataword.

- **Modification name**—If required, this is a name that specifies the digit modification string that is to be inserted beginning at the application point.

Dataword rules:

- Dataword1 must be 1 through 32 or 98.
- Dataword2 must be 0 through 32 or 99.
- Dataword3 must be 0 or an existing digit modification name.
- Dataword4 must be 0.

For example, if the application point = 1, the number of digits to remove = 5, and the modification name gives a result of 1321, then begin at the start of the digit string, remove 5 digits, and replace them with the digit string 1321. This yields the two following B-number values:

- B-number received pre-analysis = 01444 567891
- B-number post analysis = 1321 567891

For example, if the application point = 98, the number of digits to remove = 4, and the modification name gives a result of 1321, then begin at the end of the digit string, remove 4 digits, and replace them with the digit string 1321. This yields the two following B-number values:
• B-number received pre-analysis = 12345567891
• B-number post analysis = 12345561321

Depending on the analysis area that invokes it, the BMODDIG result type has different functionality. The following are examples of this different functionality:

• In Pre-Analysis there are currently four serial stages that can produce the BMODDIG result type. In Pre-analysis, the results are cumulative. For example, if the CPC stage generates a BMODDIG result type, then the B-number is modified according to the result and this modified number is then the new B-number passed as input to the next Pre-analysis stage (TMR analysis). If the TMR analysis provokes another BMODDIG result type, then it further modifies the number and so on. Even though multiple modifications like this would seem excessive and unnecessary, the capability exists to ensure that the required flexibility is provided.

• In Number analysis (A-number or B-number), functionality is different. Here digit analysis is applied (digit by digit) and it is possible to have the BMODDIG result type at multiple points if required. However, it is only the last modification result type that is applied.

---

**Note**

Digit modification is applied to the initial number input to this analysis stage. There is no cumulative digit modification performed.

For example, if the received B-number is 1234 and at “1” a BMODDIG result type is received making the number 44123 4, the digit string is modified and analysis continues according to the digit analysis configuration. If another BMODDIG result type is received at 1234, making the number 551234, the earlier BMODDIG result type (“1”) is discarded and the number now sent forward is 551234.

**B_NBR_MOD_MWI**

B_NBR_MOD_MWI result type is used to modify the B-number received on any incoming DPNSS message with NSI string for MWI. If the service indicator is set to NULL, this result is ignored. If the indicator is set to 0, then copy the B-number into the A-number and copy the digit string indexed by DW1 in the DIGMODSTRING list into the B-number. If the indicator is set to 1, then copy the B-number into the A-number and copy the digit string indexed by DW2 in the DIGMODSTRING list into the B-number. This result type is used to provision the B-number in DPNSS for MWI.

Dataword1 is the MWI ON digit modification string.

Dataword2 is the MWI OFF digit modification string.

If this result type is not configured, and the MGC receives a virtual MWI string and this service indicator is set to 0 or 1, then the call is released.

**B_NUMBER_TYPE**

The B_NUMBER_TYPE result type lets you change the A-number or B-number type NOA from that presented in the IAM or Setup message. The value given as data in the result type (dataword1) is the Cisco PGW 2200 Softswitch internal call context value for the new NOA relating to either the A-number or B-number. This result type is available to A-number analysis or B-number analysis.

---

**Note**

The NOA value needs to be the MGC internal value and not the protocol-specific value. See Appendix A, “NOA and NPI Codes, CPC and TMR Values” for specific protocol values.
BSM_REQ
The BSM_REQ result type indicates that the basic service markings (BSM) have not been supplied and are required for the outgoing side.

CALL_CUTOFF_TIMER
The CALL_CUTOFF_TIMER result type terminates any call that exceeds the preset duration of the timer. The timer value is initially read from the XECfgParm.dat file. The default value range is 0 to 48 hours, in 1 hour intervals. The timer value can also be set by dataword1 in the CALL_CUTOFF_TIMER result type.

Note
In software Release 9.5(2), dataword1 values for minutes (1–2880) and seconds (1–172800) were added, along with dataword2 (callcutoffunits).

Dataword1 is the call cutoff timer value. The value range is 0 through 2.
- 0 = 1–48 (A value of 0, default, disables the call cutoff timer.)
- 1 = 1–2880
- 2 = 1–172800

Dataword2 is the call cutoff timer units. The value range is 0 through 2.
- 0 = Hours (default)
- 1 = Minutes
- 2 = Seconds

If the timer value is set to 0 by means of dataword1 for the CALL_CUTOFF_TIMER result type, then the call cutoff timer is disabled, which takes precedence over the global timer value set in the XECfgParm.dat file for calls associated with this result type.

If the timer value is set to any other value (1 through 48 hours) by means of dataword1 in the result type, then the cutoff timer is set to this value, which also takes precedence over the global timer value set in the XECfgParm.dat file for calls associated with this result type.

If this result is not configured against the call during setup, the Cisco PGW 2200 Softswitch uses the global timer value set in the XECfgParm.dat file.

Note
In the rare event where failover occurs multiple times, and CALL_CUTOFF_TIMER is enabled, each failover causes the timer value to be re-applied to the currently active platform. As a result, the actual time for a call to be released might exceed the call cutoff timer value setting.

CALL_REPORT
To use the call reporting feature, you must use this result type to indicate that a call should be reported to the management station. When a call triggers the CALL_REPORT result type, the Cisco PGW 2200 Softswitch generates a new SNMP trap. When you configure the dial plan result, you can define a text string (for example, "Emergency"), which the Cisco PGW 2200 Softswitch can pass in the SNMP trap. Also, the Cisco PGW 2200 Softswitch can pass other call details (such as calling and called numbers) that are identified by the new MIB objects that have been added to the CISCO-TRANSPATH-MIB (tp.my file).

See the “Provisioning Call Reporting” section on page 4-68 for provisioning procedures of the call reporting feature.
CALL_TAG

The generic call tagging feature introduced this result type. You can use this result type to apply a tag list in Pre-analysis, A-number analysis, or B-number analysis on the Cisco PGW 2200 Softswitch.

Dataword1 of the CALL_TAG result type names a tag list. A tag list contains tag pairs, which are formed by a tag name and a tag value. However, a tag list can contain just a tag name with the default tag value “true”.

See the “Provisioning Generic Call Tagging” section on page 4-75 for provisioning procedures of the generic call tagging feature.

CAUSE

The Cause analysis data specifies what actions to take when a given cause code and location are presented to analysis. The cause might have been retrieved from a received message, set internally on the MGC, or delivered as a CAUSE result. Currently, the given cause value is passed into the Cause analysis process and determines whether or not to:

1. Reattempt, redirect, or reroute the call on an alternate route.
2. Return an announcement (that is, route to the announcement server).
3. Clear the call down, writing the cause value returned into call context for protocol use.

The cause code corresponds to any provisioned value that complies with the range of cause values permitted in call context. See Appendix B, “Cause and Location Codes” for cause code values.

The CAUSE result type has the following datawords:

- **Cause**
  - Valid values for this dataword are
    - 0 = No cause mapping (default).
      - The 0 value is added to enable using a wildcard for the cause value. Provides a default value for cause values not manually provisioned. Use the received cause value.
    - 1 through 173 = Cause mapping value.

- **Location**
  - Valid values for this dataword are
    - 0 = No location mapping (default). The 0 value enables a wildcard location value. Use the default location value if no location is received.
    - 1 through 13 = Location mapping value. The location value corresponds to any provisioned value that complies with the range of location values permitted in call context. See Appendix B, “Cause and Location Codes,” for location values.

CC_DIG

The CC_DIG result type retrieves and stores the Country code digits for the B-number during B-number analysis. These digits can then be used to prefix the B-number when the Cisco PGW 2200 Softswitch is functioning in a National switching node capacity.

Dataword1 provides a Modification name that the Cisco PGW 2200 Softswitch uses to read the DIGMODSTRING in the dial plan. This enables the applicable Country code digits to be provisioned in the DIGMODSTRING as any other set of number modification digits. See the “Provisioning the CC_DIG Result Type” section on page 4-58 for more detailed information.
The decision to apply the stored Country code digits as a prefix to the B-number is based on the BDigitCCPrefix property setting on the selected egress Trunk Group, which occurs after analysis. Thus, at this stage of call processing, if the BDigitCCPrefix property is set to applying the Country code prefix, then the Cisco PGW 2200 Softswitch uses previously retrieved digits (from DIGMODSTRING) to modify the B-number.

See the “Adding or Removing Country Code” section on page 1-100 for more detailed information on how to prefix the country code to an A- or B-number when the Cisco PGW 2200 Softswitch is functioning in a National switching node capacity.

**CG_PN_COPY**

The CG_PN_COPY result type copies the Calling Party number (CgPn) to the value of the presentation number parameter. This allows automatic filling of the CgPN address with the provisioned network number when the existing address digits are moved to the Generic Number-Additional calling party number (GN-ACgPN). The associated NOA, NPI, Screening Indicator (SI), and Presentation Indicator (PI) fields are copied from the calling party number to GN-ACgPN. Currently the following associated data is set in Call Context for CgPn: NOA- NAT, SI-Network Provided, and PI- Allowed. If the calling number is displayed on a called party’s phone, it is the presentation number and not the CgPn, because the result type has changed it. If dataword1 is null, then the CgPN is left intact after the existing digits are moved. PN is a historical term, although still used a lot, but the correct term is GN-ACgPN.

**Note**

The TNUP protocol variant only has a PN.

**CG_PRES_IND**

The CG_PRES_IND result type changes the presentation indicator based on number analysis. The possible values are

- 0—default
- 1—restricted
- 2—allowed
- 3—unavailable

**CG_SCREEN_IND**

The CG_SCREEN_IND result type is the screening indicator of the calling party number. The screening indicator of the calling party number is modified with this result type.

Dataword1 is the calling party number screening indicator value. The screening indicator values are

- 1—NP (network provided)
- 2—UPVP (user provided verified and passed)
- 3—UPNV (user provided not verified)
- 4—UPVF (user provided verified and failed)
- 5—spare1

**CHARGE**

The CHARGE result type provides charging information relevant to the call and it supports the German, India, and Polish Advice of Charge (AOC) functionality (shown in Table 1-3) as determined by the ingress trunk group property **AOC Enabled**. Number analysis is responsible for obtaining the Charge
Origin and Charge Destination information from the dial plan and passing this information to the CDR Manager, where it is used to access the Charge values. The information fields retrieved from the CHARGE result type are defined as follows:

- **Charge Data Discriminator**—Determines the type of data in dataword1. Values are as follows:
  - **1 = Tariff Rate**—Used when the tariff rate is fixed and is independent of origin or time.
  - **2 = Charge destination**—Used for origin and/or time dependent tariff rates for customers requiring this capability due to inter-operability agreements or certification requirements.
  - **3 = Charge Band**—Used for AOC to generate a charge band number to the preceding exchange in the SS7 network as part of the charge message. When an originating switch that is to charge the call receives this value, it is used as an index into the charging table being used to calculate the charge amount for the call; or to start charging the call based on the value derived from the charging table.
  - **4 = Charge Unit**—When an originating switch that is supposed to charge the call receives this value (0-255), it starts charging the call based on this charge unit value. For example, a number of seconds is associated with each charge unit (or charge rate); thus the call duration after the answer signal is divided into charge units until the end of the call. The charge units are then converted into a monetary value and the user is billed accordingly.
  - **5 = Meter Pulse**—Indicates that the meter pulse table is read instead of the tariff table with the tariff descriptor value obtained from Charge table reading.

- **Scale Factor**—Determines the value that corresponds to a multiplication factor (see Table 1-2) that is applied to the tariff rate. Set to 1 for metering pulses.

**Table 1-2 Tariff Rate Scale Factor Values**

<table>
<thead>
<tr>
<th>Value</th>
<th>Scale Factor</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>x 1000</td>
</tr>
<tr>
<td>2</td>
<td>x 100</td>
</tr>
<tr>
<td>1</td>
<td>x 10</td>
</tr>
<tr>
<td>0</td>
<td>x 1</td>
</tr>
<tr>
<td>255</td>
<td>x 0.1</td>
</tr>
<tr>
<td>254</td>
<td>x 0.01</td>
</tr>
<tr>
<td>253</td>
<td>x 0.001</td>
</tr>
<tr>
<td>252</td>
<td>x 0.0001</td>
</tr>
<tr>
<td>251</td>
<td>x 0.00001</td>
</tr>
<tr>
<td>250</td>
<td>x 0.000001</td>
</tr>
<tr>
<td>249</td>
<td>x 0.000001</td>
</tr>
</tbody>
</table>

- **Charge Type**—Set to 1 (for German AOC) or to 0 (for India and Polish AOC) to indicate AOC. This value is determined by the selected protocol variant.

The result data is returned only when analysis and routing are completed. For the Cisco PGW 2200 Softswitch, this is when a trunk group is returned for circuit selection.
Table 1-3 gives definitions of the charge result type.

### Table 1-3  **CHARGE Result Type Definitions**

<table>
<thead>
<tr>
<th>Result Type</th>
<th>Protocol Variant</th>
<th>Dataword1</th>
<th>Dataword2</th>
<th>Dataword3</th>
<th>Dataword4</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHARGE</td>
<td></td>
<td>TariffRate/ Dest for Charging/ Charge Band Number/ Charge Unit</td>
<td>Scale Factor</td>
<td>ChargeData Discriminator</td>
<td>Charge Type</td>
</tr>
<tr>
<td>German</td>
<td></td>
<td>0-999999</td>
<td>0-3, 249-255</td>
<td>1 (Tariff Rate)</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>1-9999</td>
<td>0</td>
<td>2 (Charge Destination)</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>India</td>
<td>0-255</td>
<td>0</td>
<td>3 (Charge Band)</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Polish ISUP V2</td>
<td>0-255</td>
<td>0</td>
<td>3 (Charge Band)</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Finnish</td>
<td>0-999999</td>
<td>0</td>
<td>5 (Meter Pulse)</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

**Note**

When provisioning the CHARGE result type, use the values shown in Table 1-3 for the protocol variant you are using.

**CHARGE_IND**

The CHARGE_IND result type (Charge indication) indicates whether the Cisco PGW 2200 Softswitch should change the value of the charge indicator. The CHARGE_IND result type can be provisioned for A-number and B-number analysis and is an intermediate result.

**ChargeModeInd**—This dataword (dataword1) has the following values:

- 0 = Leave as it is (default)
- 1 = Charge
- 2 = No charge

**CHARGE_MODE_IND**

The Charge Mode Indicator (CHARGE_MODE_IND) result type indicates how the metering pulses generated by the MGC are applied in relation to possible other metering pulses (generated by some other node). The CHARGE_MODE_IND result type is assignable against the ADIGTREE or BDIGTREE component and is an intermediate result.

**ChargeModeInd**—This dataword (dw1) has the following values:

- 1 = Add on charge
- 2 = Replace charge
- 3 = Free of charge

**CHARGEORIGIN**

The CHARGEORIGIN result type contains an integer value in the range of 1–9999 and is returned during A-number analysis if the Advice of Charge feature is enabled on the ingress trunk group or sigpath.

The charge origin value is determined in one of three ways:

- From the charge origin value ACHGORIGIN
- From the A-number result type CHARGEOIGIN
- From the trunk group/sigpath property ChargeOrigin

The Charge Origin value (ACHGORIGIN) takes precedence over the CHARGEOIGIN result returned from A-number analysis, which takes precedence over the charge origin property (ChargeOrigin) defined against the trunk group or sigpath.

If none of these three options provides a charge origin value, and AOC is enabled, the charge origin value defaults to 0.

**Note**
For the CPC_REQ, CLI_REQ, BSM_REQ, and FSM_REQ result types, the required information can be retrieved by an internal request signal if the originating protocol supports backward requests. If the protocol does not support such requests, the call progresses without this information and the next exchange determines if it is required.

**CLI_NBR_LENGTH**
The CLI_NUMBER_LENGTH result type basically indicates that the calling line identity has the incorrect number of digits. The Numbering Type field is not processed, but the maximum and minimum digit fields are used to determine whether the CLI is too long or too short. If the CLI is too long or too short, a negative result is returned, the cause is set to IC_BLACKLIST_CLI_LENGTH_INVALID, and the call is released. The protocol can apply a different cause code in the outgoing release message.

**CLI_REQ**
The CLI_REQ result type indicates that the calling line identity (CLI) has not been supplied and is required for the outgoing side.

**CODEC**
The CODEC result type indicates the codec support required for an incoming message. Dataword1 indicates the codec string name used by the result and dataword2 indicates if the codec action is mandatory (0) or preferred (1). Dataword3 indicates the type of the codec string that Dataword1 contains:

1 = Indicates that the codec string in dataword1 is an audio codec string.
2 = Indicates that the codec string in dataword1 is a video codec string.

**COND_ROUTE**
This result should be configured only when time conditional routing is required. When this result type is returned, the Cisco PGW 2200 Softswitch prepares data and enters the Conditional Routing analysis stage.

When the COND_ROUTE result type is added, the user configures the CondRouteName. The result is added with the start name in dataword1. The dataword CondRouteName is also one of the access keys used to read the Day/Time data associated with this result from the condRoute value in the Routing data file.

For more information see the “Conditional Route Description” section on page 1-114.
CPCMOD
In A-number analysis, you can use the CPCMOD result type to modify the CPC of the IAM message to include the desired indicator. For example, the Cisco PGW 2200 Softswitch A-number can be provisioned with a list of the numbers that are configured as payphones. A-number analysis then handles calls from these numbers by returning the CPCMOD result type with the payphone indicator (0xF) set in dataword1. This result type is then used to modify the CPC information in the IAM message. When calls from payphones are routed to the PSTN or other carriers, the CPC information in the IAM message indicates that the call originated from a payphone so the proper billing information is provided.

Note
The CPC value needs to be the MGC internal value and not the protocol-specific value.

CPC_REQ
The CPC_REQ result type indicates that the calling party category (CPC) has not been supplied and is required for the outgoing side.

DATA_EXCHANGE
The DATA_EXCHANGE result type delivers a result from B-number analysis indicating that there are actions required to move certain data from one call context location to another. For example, if the result indicates a home-based local routing number (LRN), then the B-number and the generic address parameter (GAP) number must be exchanged, and then new B-number analysis is invoked. The entry in the associated ActionType field indicates the type of action that is required. Currently the only value is 1 = Home LRN—This number is a home LRN, that is, local to this Cisco PGW 2200 Softswitch. This signifies that the Cisco PGW 2200 Softswitch must complete the call to the dialed number contained in the GAP (not the number in the B-number). Consequently the GAP and B-numbers must be exchanged.

DB_XLATED
The DB_XLATED result type provides database look up and number translation for both ported and non-ported types of calls. The DB_XLATED result type also allows you to change the dial plan based on a matched or non-matched database query. This removes the previous requirement (by means of the E_PORTED_NUM result type) to provision a default ROUTE result, which was used in the event that the database query failed to find a match. However, if no dial plan options are configured in the DB_XLATED result (dataword2 and dataword3), a default ROUTE or NEW_DIALPLAN result is still necessary.

This result type has the following data words:
- searchMin—Value indicating how far to search back in the number when longest matching.
- matchNewDp—Entry index (integer) to a new dial plan in the dial plan selection table for Cisco PGW 2200 Softswitch to switch to for further processing following a database reading indicating that the target was matched. The dataword is provisioned as a dial plan name. It is then internally converted to an integer value to point to an entry in the dial plan selection table.
- nonMatchedNewDp—Entry index (integer) to a new dial plan in the dial plan selection table for Cisco PGW 2200 Softswitch to switch to for further processing following a database reading indicating the target was not matched. The dataword is provisioned as a dial plan name. It is then internally converted to an integer value to point to an entry in the dial plan selection table.

The following items further describe the behavior of the DB_XLATED result type:
- It is possible to collect DB_XLATED at any point in B-number analysis and always issue a database query regardless of any later ROUTE or other final result.
• For enbloc calls or for overlap calls where sending is complete, a longest matching database query is made.

• For overlap calls without a ST digit present, the Cisco PGW 2200 Softswitch performs a partial matching query.

• For overlap calls without a ST digit present, if the initial partial matching database query finds no matches, the Cisco PGW 2200 Softswitch launches a new query using longest matching.

• Result types DB_XLATED, E_PORTED_NUM, E_ROUTE_NUM, and TERM_INFO are all mutually exclusive; the last one collected is the one processed.

• Multiple retrievals of the DB_XLATED result also mean that the last DB_XLATED result is the result that is processed.

• The existing E_PORTED_NUM result is unchanged and provides the current level of LNP-only functionality.

• The DB_XLATED result provides database lookup and number translation for ported and non-ported calls.

• Both the E_PORTED_NUM and DB_XLATED result types query the ported number table, but use different methods to read the table.

**DEFAULT_TMR**

The DEFAULT_TMR result type allows the Cisco PGW 2200 Softswitch to set or overwrite the TMR value.

Dataword1 specifies the TMR value for this call:

- 1 = Set the TMR value to SPEECH.
- 2 = Set the TMR value to UNRES_64K.
- 3 = Set the TMR value to AUDIO_3K.

In the following example, you overwrite the TMR value for all of the calls whose calling numbers start with 400. The TMR values for these calls are set to unrestricted 64k (UNRES_64K).

```
numan-add:resultset:custgrpid="1111",name="sip-tmr"
numan-add:resulttable:custgrpid="1111",resulttype="DEFAULT_TMR",dw1="2",setname="sip-tmr",name="tmrdata"
numan-add:adigtree:custgrpid="1111",callside="originating",setname="sip-tmr",digitstring="400"
```

**DIGIT_REQ**

The DIGIT_REQ result type indicates that insufficient digits were received for analysis to provide a result with which call processing can be continued. This result type returns an indication to the call module of how many more digits are required for analysis to be completed by subtracting the number of digits returned in the analysis result type from the number of digits that have already been received.

**Note**

This result type is for use with overlap signaling. Thus this result might not be initiated if the protocol receiving it does not support overlap signaling.
**DRP_EXIT**

The DRP_EXIT result type directs the Cisco PGW 2200 Softswitch to exit from the DRP stage of preanalysis. For more information on provisioning procedures, see “Provisioning Domain Based Routing” section on page 4-73.

DRP_EXIT has dataword1, drpExitType. DW1 drpExitTyp e specifies the type of DRP exit. Valid values are:

- 1 = Directs the Cisco PGW 2200 Softswitch to exit current DRP Step and move to the next step.
- 2 = Directs the Cisco PGW 2200 Softswitch to exit from entire DRP stage of pre-analysis.

**DTMFCAP**

The DTMF result type is returned from A-number or B-number analysis indicating the DTMF capability of the number in the dial plan. This result type can be encountered during A-number or B-number analysis and indicates the DTMF capability of the associated number in the dial plan. DTMF capability on B-number analysis overrides DTMF capability on A-number analysis.

Dataword1 defines the capability of egress trunk group. The value range is 0 through 2.

- 0—Ignore DTMF capability
- 1—RFC 2833 DTMF capability
- 2—Out of band DTMF capability

**E_PORTED_NUM**

The E_PORTED_NUM result type is an indication to read the ported number data. The ported number data can only be read if all digits have been received. Thus in enbloc processing can continue directly; in overlap the Cisco PGW 2200 Softswitch must wait until sending is complete. The ported result can be provisioned at the area code level, but the ported number is not accessed until the complete number is received. See the “European Local Number Portability” section on page 1-88 for more information.

**Note**

The E_PORTEDNUM result type is provisioned only when the Cisco PGW 2200 Softswitch is in the donor switch capacity for European LNP.

The E_PORTED_NUM result type has the following data words:

- **Number of Prefix Digits to remove before reading the ported number data**—Indicates the number of prefix digits that must be removed from the number (1 through 32, the default is 0). This option provides flexibility by enabling any normalization prefix digits to be removed before the Cisco PGW 2200 Softswitch prefixes the routing number.

- **UsePartialNumber**—Indicates whether the Cisco PGW 2200 Softswitch interrogates the TimesTen database with a full or partial number. This dataword has the following values:
  - 0 = Full number (default). This value forces enbloc behavior.
  - 1= Partial number.

**E_ROUTE_NUM**

The E_ROUTE_NUM result type indicates that the Cisco PGW 2200 Softswitch must remove the routing number prefixing the Called Number, then access the number termination table to get a route list name with which to route the call. See the “European Local Number Portability” section on page 1-88 for more information.
Note  The E_ROUTE_NUM result type is provisioned only when the Cisco PGW 2200 Softswitch is in the recipient switch capacity for European LNP.

The E_ROUTE_NUM result type has the following datawords:

- **RemovePfxDig**—Integer value indicating the number of prefix digits to remove from called number before the Cisco PGW 2200 Softswitch reads the Ported or Number termination database table. The default value is 0.

- **UsePartialNumber**—Indicates whether the Cisco PGW 2200 Softswitch interrogates the TimesTen database with a full or partial number. This dataword has the following values:
  - 0 = Full number (default). This value forces enbloc behavior.
  - 1 = Partial number.

**E911PROF**

The E911PROF result type is returned from B-number analysis (the called number) indicating if the B-number is an emergency call and the profile mapping to apply to emergency numbers.

Valid E911PROF dataword1 values are listed in Table 1-4.

### Table 1-4 E911PROF Dataword1 Result Type Mapping

<table>
<thead>
<tr>
<th>Dataword 1</th>
<th>ISUP Parameter Option</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ESRK Delivery&lt;sup&gt;1&lt;/sup&gt;</td>
</tr>
<tr>
<td>1</td>
<td>A1</td>
</tr>
<tr>
<td>2</td>
<td>A2</td>
</tr>
<tr>
<td>3</td>
<td>A3</td>
</tr>
<tr>
<td>4</td>
<td>B1</td>
</tr>
<tr>
<td>5</td>
<td>B2</td>
</tr>
<tr>
<td>6</td>
<td>B3</td>
</tr>
<tr>
<td>7</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
</tr>
</tbody>
</table>
Table 1-4  E911PROF Dataword1 Result Type Mapping (continued)

<table>
<thead>
<tr>
<th>Dataword 1</th>
<th>ISUP Parameter Option</th>
<th>ESRK Delivery(^1)</th>
<th>CBN and ESRD Delivery(^2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>21</td>
<td></td>
<td>F1</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td></td>
<td>F2</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td></td>
<td>G1</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td></td>
<td>G2</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td></td>
<td>H1</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td></td>
<td>H2</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td></td>
<td>I1</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td></td>
<td>I2</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td></td>
<td>I3</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td></td>
<td>J1</td>
<td></td>
</tr>
<tr>
<td>31</td>
<td></td>
<td>J2</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td></td>
<td>K1</td>
<td></td>
</tr>
<tr>
<td>33</td>
<td></td>
<td>K2</td>
<td></td>
</tr>
<tr>
<td>34</td>
<td></td>
<td>L1</td>
<td></td>
</tr>
<tr>
<td>35</td>
<td></td>
<td>L2</td>
<td></td>
</tr>
</tbody>
</table>


FACILITY

The FACILITY result allows you to

- control redirection and call transfer behavior for originating and terminating devices.
- set SIP call handling to proxy mode for an individual call.

Note  This capability requires that the sipModeSelectionControl parameter be set to permit B2BUA or Proxy mode.

- observe how the Cisco PGW 2200 Softswitch handles redirection and SIP Refer.

Note  You can configure FACILITY for source (A-number) or destination (B-number).

- reject calls conditionally based on source or destination domain name.
The Facility result type has the following datawords:

- **Type**—Controls handling of Call Transfer and Refer requests. Valid values:
  - 1 = Proxy Mode required
  - 2 = Originating Redirection treatment action
  - 3 = Originating Call Transfer treatment action
  - 4 = Terminating Redirection treatment action
  - 5 = Originating Redirection Rejection treatment action
  - 6 = Terminating Call Transfer treatment action

- **Treatment**—Determines the actions required based on the Type dataword value. Valid values:
  - 1 = Not supported
  - 2 = Always supported
  - 3 = Supported conditionally upon matching domain
  - 4 = Supported conditionally upon non-matching domain
  - 5 = Unconditional rejection of Terminating Redirection/Call transfer Request
  - 6 = Conditional rejection (if Non-E164) of Terminating Redirection Request/Call Transfer request

**Note**
Value 5 and 6 determine the action taken when the terminating side of a call issues a redirect.

**Note**
If Type is set to Proxy Mode, Treatment is not used.

The Cisco PGW 2200 Softswitch can allow a SIP REFER on the terminating side of the call to be propagated back to the originating side (SS7) of the call by sending a REL message containing the redirection number and redirection information. You can enable this service by provisioning a FACILITY result with DW1 set to 3 and DW2 set to 2.

*Table 1-5* provides the call processing treatment applied according to the combinations of parameter sipModeSelectionControl and the dataword values from the FACILITY result type. Unless otherwise stated, sipModeSelectionControl is set to value 1 (b2bua optional).
**Table 1-5  SIP and Non-SIP Call Processing Actions According to FACILITY Configuration**

<table>
<thead>
<tr>
<th>Dataword1 Value</th>
<th>Dataword2 Value</th>
<th>Call Processing Action on the Cisco PGW 2200 Softswitch</th>
</tr>
</thead>
</table>
| 1 Any.          | 1 (Backward transit of redirection to originating side not allowed.) | Action is according to the value of the XECfgParm parameter sipModeSelectionControl:  
  - If sipModeSelectionControl=2, then the result-type is ignored because the main parameter is set for proxy mode.  
  - If sipModeSelectionControl=1, then set the Cisco PGW 2200 Softswitch to indicate that Proxy mode is required for this call.  

| 2 (Originating Redirection treatment action) | 2 (Backward transit of redirection to originating side is supported unconditionally.) | This combination of dw1 and dw2 sets the originating side redirection action to indicate that backward transit of a redirection is *not supported* on the originating side of the Cisco PGW 2200 Softswitch.  
When Cisco PGW 2200 Softswitch call control receives a redirection request from the Cisco PGW 2200 Softswitch terminating side, it does not try to send back the redirection to the preceding switch. The existing local handling of redirection (that is, using cause analysis) applies.  
(Applicable to SIP, DPNSS, and QSIG.)  

| 2 (Originating Redirection treatment action) | 3 (Backward transit of redirection to originating side is conditionally supported on matching domains.) | This combination of dw1 and dw2 is appropriate for a SIP B2BUA call (that is, SIP originating and SIP terminating).  
If Cisco PGW 2200 Softswitch call control receives a redirection request from the SIP terminating side, it transits the request back to the SIP originating side for sending out to the preceding switch. The only limitation is if the Cisco PGW 2200 Softswitch originating side protocol cannot support this handling.  
(Applicable to SIP, DPNSS, and QSIG.)  

| 2 (Originating Redirection treatment action) | 1 (Backward transit of redirection to originating side) | This combination of dw1 and dw2 sets the originating side redirection action to indicate that backward transit of a redirection is *supported* on the originating side of the Cisco PGW 2200 Softswitch.  
If Cisco PGW 2200 Softswitch call control receives a redirection request from the terminating side, it transits the request back to the originating side for sending out to the preceding switch. The only limitation is if the Cisco PGW 2200 Softswitch originating side protocol cannot support this handling.  
(Applicable to SIP, DPNSS, and QSIG.)  

| 2 (Originating Redirection treatment action) | 3 (Backward transit of redirection to originating side) | This combination of dw1 and dw2 is appropriate for a SIP B2BUA call (that is, SIP originating and SIP terminating).  
If Cisco PGW 2200 Softswitch call control receives a redirection request from the SIP terminating side, it transits the request back to the SIP originating side for sending out to the preceding network entity. This happens only if the domain in the From header received within the original INVITE on the OCC side matches the domain received within the Contact header received back in the 302 message on the SIP terminating side.  
The redirection is transited back if the required domain of the redirected destination is the same as that of the originator of this call.  
(Applicable to SIP.)
### Table 1-5  SIP and Non-SIP Call Processing Actions According to FACILITY Configuration (continued)

<table>
<thead>
<tr>
<th>Dataword1 Value</th>
<th>Dataword2 Value</th>
<th>Call Processing Action on the Cisco PGW 2200 Softswitch</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 (Originating Redirection treatment action)</td>
<td>4 (Backward transit of redirection to originating side is conditionally supported on nonmatching domains.)</td>
<td>This combination of dw1 and dw2 is appropriate for a SIP-originated call and can be either a B2BUA mode call or an interworking call. If Cisco PGW 2200 Softswitch call control receives a redirection request from the Cisco PGW 2200 Softswitch terminating side, it transits this back to the originating side for sending out to the preceding switch, provided that the domain received within the Contact header received back in the 302 message (terminating side) does not match the Cisco PGW 2200 Softswitch domain. In an interworking call, this provision is met because the Contact header domain is absent from the terminating side. If the call is SIP B2BUA, the provision is subject to the check as described. The redirection is transited back if the required domain of the redirected destination is not the Cisco PGW 2200 Softswitch domain. Otherwise, the Cisco PGW 2200 Softswitch can deal with this redirection locally. (Applicable to SIP originating side.)</td>
</tr>
<tr>
<td>3 (Originating Call Transfer treatment action)</td>
<td>1 (Backward transit of call transfer to Originating side is not allowed.)</td>
<td>This combination of dw1 and dw2 sets the originating side call transfer action to indicate that backward transit is not supported on the originating side of the Cisco PGW 2200 Softswitch. When Cisco PGW 2200 Softswitch call control receives a call transfer request from the Cisco PGW 2200 Softswitch terminating side, it does not try to send this back to the preceding switch. The local handling of call transfer is invoked. (Applicable to SIP and QSIG terminating side.)</td>
</tr>
<tr>
<td>3 (Originating Call Transfer treatment action)</td>
<td>2 (Backward transit of call transfer to originating side is supported unconditionally.)</td>
<td>This combination of dw1 and dw2 sets the originating side call transfer action to indicate that backward transit of a call transfer request is supported on the originating side of the Cisco PGW 2200 Softswitch. If Cisco PGW 2200 Softswitch call control receives a call transfer request from the terminating side, it transits the request back to the originating side for sending out to the preceding switch. The only limitation on this is if the Cisco PGW 2200 Softswitch originating side protocol cannot support this handling. (Applicable to SIP and QSIG.)</td>
</tr>
<tr>
<td>3 (Originating Call Transfer treatment action)</td>
<td>3 (Backward transit of call transfer to originating side is conditionally supported on matching domains.)</td>
<td>This combination of dw1 and dw2 is appropriate for a SIP originated B2BUA mode call where REFER actions have been requested on the Cisco PGW 2200 Softswitch terminating side. With this setting, the backward transit of a REFER request is conditionally supported on the originating side of the Cisco PGW 2200 Softswitch. When the Cisco PGW 2200 Softswitch terminating SIP side receives a REFER request and passes the request back to call control, Cisco PGW 2200 Softswitch call control transits this request back to the Cisco PGW 2200 Softswitch originating side provided that the received Refer-To header domain in the REFER message (terminating side) matches the domain in the From header received within the original INVITE on the OCC side. The REFER back is transited if the required domain of the refer-to destination is the same as the originator of this call. (Applicable to SIP.)</td>
</tr>
</tbody>
</table>
### Table 1-5  **SIP and Non-SIP Call Processing Actions According to FACILITY Configuration (continued)**

<table>
<thead>
<tr>
<th>Dataword1 Value</th>
<th>Dataword2 Value</th>
<th>Call Processing Action on the Cisco PGW 2200 Softswitch</th>
</tr>
</thead>
</table>
| 3 (Originating Call Transfer action) | 4 (Backward transit of call transfer to originating side is conditionally supported on nonmatching domains.) | This combination of dw1 and dw2 is appropriate for a SIP originated B2BUA mode call where REFER actions have been requested on the Cisco PGW 2200 Softswitch terminating side.  
With this setting, the backward transit of a REFER request is conditionally supported on the originating side of the Cisco PGW 2200 Softswitch. When the Cisco PGW 2200 Softswitch terminating SIP side receives a REFER request and passes this request back to call control, Cisco PGW 2200 Softswitch call control transits this request back to the Cisco PGW 2200 Softswitch originating side provided that the received refer-to header domain in the REFER message (terminating side) does not match the Cisco PGW 2200 Softswitch domain.  
The REFER back transits if the required domain of the refer-to destination is not the Cisco PGW 2200 Softswitch domain. Otherwise, the Cisco PGW 2200 Softswitch can deal with this locally.  
(Applicable to SIP.) |
| 4 (Terminating Redirection treatment action) | 1 (Unconditional SIP recursion.) | This combination of dw1 and dw2 is specific to a SIP-terminated call and is designed to invoke SIP recursive redirection handling.  
However, with this direct combination there is an inherent risk of looping. To avoid looping, the actual behavior associated with this combination is the same as the combination of dw1=4 and dw2=3.  
(Applicable to SIP.) |
| 4 (Terminating Redirection treatment action) | 2 (Unconditional passing of redirection request back to call control.) | This combination of dw1 and dw2 is appropriate for the receipt of a redirection request on the Cisco PGW 2200 Softswitch terminating side.  
In this situation, the terminating side checks the FACILITY setting for the appropriate call processing action. The value 2 indicates that this request can be passed back to Cisco PGW 2200 Softswitch call control for further handling.  
**Note**  
The actual transit of the request back out on the Cisco PGW 2200 Softswitch originating side depends on the FACILITY setting for that side.  
(Applicable to SIP, DPNSS, QSIG, and SS7.) |
| 4 (Terminating Redirection treatment action) | 3 (Conditional passing of redirection request back to call control on matching domains.) | This combination of dw1 and dw2 is appropriate for a SIP-terminated call where the Cisco PGW 2200 Softswitch SIP terminating side on receipt of a 3xx response provoking a redirection, checks the FACILITY setting for the appropriate call processing action.  
The value 3 indicates that the request can be passed back into call control provided that the domain in the received Contact header within the 3xx message matches the Cisco PGW 2200 Softswitch domain. If this is the case, then the Cisco PGW 2200 Softswitch terminating side passes this request back to Cisco PGW 2200 Softswitch call control for further handling.  
**Note**  
The actual transit of the request back out on the Cisco PGW 2200 Softswitch originating side depends on the FACILITY setting for that side.  
The redirection request transits back to call control if the required domain of the redirected destination is the Cisco PGW 2200 Softswitch domain where the Cisco PGW 2200 Softswitch can deal with this request locally. If this is not the case, then SIP recursion is used. |
### Table 1-5  
**SIP and Non-SIP Call Processing Actions According to FACILITY Configuration (continued)**

<table>
<thead>
<tr>
<th>Dataword1 Value</th>
<th>Dataword2 Value</th>
<th>Call Processing Action on the Cisco PGW 2200 Softswitch</th>
</tr>
</thead>
<tbody>
<tr>
<td>4 (Terminating Redirection treatment action)</td>
<td>4 (Conditional passing of redirection request back to call control on nonmatching domains.)</td>
<td>This combination of dw1 and dw2 is appropriate for a SIP terminated call where the Cisco PGW 2200 Softswitch SIP terminating side, on receipt of a 3xx response provoking a redirection, checks the FACILITY setting for the appropriate call processing action. The value 4 indicates that the request can be passed back into call control if the domain in the received Contact header within the 3xx message does not match the domain in the To header sent in the outgoing INVITE (and received back in the 3xx message). If this is the case, then the Cisco PGW 2200 Softswitch terminating side passes this request back to Cisco PGW 2200 Softswitch call control for further handling. <strong>Note</strong> The actual transit of the request back out on the Cisco PGW 2200 Softswitch originating side depends the FACILITY setting for that side. The redirection request passes back to call control if the required domain of the redirected destination is not the same domain as that previously attempted in the outgoing INVITE. If the domains are the same, then SIP recursion can be used (sending the new request out to the same domain it is already set up to use).</td>
</tr>
<tr>
<td>4 (Terminating Redirection treatment action)</td>
<td>5 (Unconditional rejection of Terminating Redirection/Call transfer Request)</td>
<td>Unconditional rejection of Redirection Request (SIP 302).</td>
</tr>
<tr>
<td>4 (Terminating Redirection treatment action)</td>
<td>6 (Conditional rejection (if Non-E164) of Terminating Redirection Request/Call Transfer request)</td>
<td>Rejection of Redirection Request (SIP 302) when the CONTACT header is non-E.164.</td>
</tr>
<tr>
<td>5 (Originating Redirection Rejection treatment action)</td>
<td>1</td>
<td>This combination of dw1 and dw2 is relevant only when the redirection request has been transmitted to the originating side. This combination determines how a rejection for the request should be handled. A value 1 for dw2 means that if Cisco PGW 2200 Softswitch call control receives a REJECT from the originating side in response to a redirection request, it transits the REJECT to the terminating (that is, requesting) side. (Applicable only to QSIG-QSIG calls.)</td>
</tr>
<tr>
<td>5 (Originating Redirection Rejection treatment action)</td>
<td>2</td>
<td>This combination of dw1 and dw2 is relevant only when the redirection request has been transmitted to the originating side and determines how a rejection for the request should be handled. A value 2 for dw2 means that if Cisco PGW 2200 Softswitch call control receives a REJECT from the originating side in response to a redirection request, call control attempts to handle the redirection request locally by invoking cause analysis. <strong>Note</strong> This is the default behavior in the absence of a provisioned originating redirection rejection treatment action.</td>
</tr>
</tbody>
</table>
Table 1-5  **SIP and Non-SIP Call Processing Actions According to FACILITY Configuration (continued)**

<table>
<thead>
<tr>
<th>Dataword1 Value</th>
<th>Dataword2 Value</th>
<th>Call Processing Action on the Cisco PGW 2200 Softswitch</th>
</tr>
</thead>
<tbody>
<tr>
<td>6 (Terminating Call Transfer treatment action)</td>
<td>5 (Unconditional rejection of Terminating Redirection /Call transfer Request)</td>
<td>Unconditional rejection of Call Transfer/Refer requests.</td>
</tr>
<tr>
<td>6 (Terminating Call Transfer treatment action)</td>
<td>6 (Conditional rejection (if Non-E164) of Terminating Redirection Request/Call Transfer request)</td>
<td>Rejection of Call Transfer/Refer Requests when the Refer-To header is non-E.164.</td>
</tr>
</tbody>
</table>
| none | none | Default behavior  
On the Cisco PGW 2200 Softswitch terminating side, the redirection or call transfer request behavior defaults to passing the request back to call control where it can be handled locally or, if there is an originating FACILITY result, propagated backwards to the previous network entity.  
On Cisco PGW 2200 Softswitch Originating side, the behavior defaults to local handling by call control and cause analysis or half-call handling rather than transiting the request back out on the Originating side. |

**FSM_REQ**

The FSM_REQ result type indicates that the facility service markings (FSM) have not been supplied and are required for the outgoing side.

**GATEWAYPOOL**

The GATEWAYPOOL result type enables the Cisco PGW 2200 Softswitch to override the GatewayPool and AnchorMedia properties provisioned on the ingress or the egress trunk group. The GATEWAYPOOL result type can be set on A number analysis or B number analysis. The B number analysis has greater priority than A number analysis when a result type such as GATEWAYPOOL is provisioned on both A number analysis and B number analysis. Specifically, if AnchorMedia is set to Never on the ingress and egress sides in the dialplan, no media anchoring operates on ingress and egress call legs.

**HLCMOD**

The HLCMOD result type allows you to modify the High Layer Compatibility of outgoing Initial Address Messages (IAMs) based on the dialed Called Party Number. You can provision this result type using A and B number analysis.

The HLCMOD result type has the following datawords:

- **HLC name**—The name of the High Layer Compatibility, such as “fax-hlc01.”
IN_SERVICE_KEY

The IN_SERVICE_KEY result type permits the assigning of a service key value according to the B-number. This result type allows multiple service keys, with each service key assigned according to the B-number. Dataword1 (any 32-bit integer value, with 0 allowed) is used to provision the IN service key used when IN triggering is initiated toward the SCP.

If multiple service keys are required, then the IN_SERVICE_KEY result type must be configured in the B-digit tree, along with the IN_TRIGGER result type. This means the IN_SERVICE_KEY result type must be provisioned into the same result-set as the IN_TRIGGER. If the single service key solution is adequate, then configure only an IN_TRIGGER result type. The IN_SERVICE_KEY result does not require configuring.

The IN_SERVICE_KEY result type has the following datawords:

- **IN Service Key**—Any Integer value including 0.
- **Global Title Digits Type**—A string representing the type of the global title digits. Valid values are:
  - CALLED
  - CALLING
  - FIXED
- **Digits name**—Name of the digit modification entry. Provision this dataword as follows:
  - If DW2 is set to FIXED, use the `numan-add:digmodstring` command to build a fixed-digit modification table and set the value of DW3 as name of the modification table.
  - If DW2 is set to CALLED or CALLING, do not provision DW3.
  - If DW2 is set to FIXED, you must provision DW3.

IN_TRIGGER

The IN_TRIGGER result type delivers a result from B-number analysis, which indicates that further analysis by an SCP is required due to an intelligent network (IN) call. The data provided identifies the service required (such as LNP) and, if necessary, an SCP/STP name for use when the TCAP call is made.

- **Service Type**—This returned value is provisioned in an internal file used to configure the handling of IN requests by the trigger module. The value returned is not processed within analysis, but is retrieved and passed back to the call module for action. This value is an indication of the type of IN service that needs to be invoked to advance this call (LNP, 800, 900, and so on). The valid Service Type values are contained in inService.dat. Valid values are
  - 0 = IN_NONE
  - 1 = IN_LNP
  - 2 = IN_800
  - 3 = ROUTE
  - 4 = IN_PLAYANN
  - 5 = IN_RELEASE
  - 6 = INPREPAID
  - 30 = IN_CNAM
- **SCP/STP Index**—Value used in the trigger module for selection of the SCP for TCAP query.
- **Minimum Digits Required**—The minimum number of digits (0 through 32) required to be received for further analysis.
Timer—The timer value (1 through 30, in seconds) used to delay the triggering if required. This timer can be started when the min digits required (dw3) are received.

**INC_NUMBERING**
The INC_NUMBERING result type returns information regarding the incoming trunk group side (OCC). This information sets the numbering criteria (overlap or en bloc) and the minimum and maximum numbers of digits permitted for the incoming trunk group side.

- **Numbering Type**—0 = Closed numbering (en bloc) or 1 = Open numbering (overlap).
- **Minimum and Maximum digits**—Refers to the minimum and maximum number lengths. In the case of closed numbering (en bloc), these values should be equal.

The data returned in this result type are used to overwrite default values loaded into the OCC at startup.

**IP_SOURCE_SCREEN**
The IP_SOURCE_SCREEN result type provides screening capabilities for non-E.164 calls. This result is supported for blacklist screening only.

For more information on provisioning procedures, see “Provisioning Domain Based Routing” section on page 4-73.

**Note**

Dataword2, dataword3, and dataword4 are optional.

**IP_DEST_TRANS**
The IP_DEST_TRANS result type translates a destination into another format, such as an E.164 destination (domain) to a non-E.164 destination (phone number). You can also use IP_DEST_TRANS to translate a non-E.164 destination to another non-E.164 destination (a domain name to another domain name). It can do the following translations:

- A domain to a phone number: contactname@cisco.com translates to 1234567890@cisco.com.
- A phone number to a domain: 1234567890@cisco.com translates to contactname@cisco.com.
- A domain to another domain: contactname@cisco.com translates to contactname@example.com.

For more information on provisioning procedures, see “Provisioning Domain Based Routing” section on page 4-73.

**IP_DEST_TRANS** has the following data words:

- **inputAndAction** (dw1)—Determines whether the Cisco PGW 2200 Softswitch translates the destination of the user and host (1) or the destination host only (2).
- serviceName (dw2)—The name of the service.
- foundSetName (dw3)—The result set that the Cisco PGW 2200 Softswitch executes if the user or domain name matches an entry in the table.
- notFoundSetName (dw4)—The result set that the Cisco PGW 2200 Softswitch executes if the user or domain name does not match an entry in the table.

**IP_ROUTE_SEL**

The IP_ROUTE_SEL result type allows the Cisco PGW 2200 Softswitch to select a route based on a destination user or domain name, source user or domain name, or a combination of the two.
For more information on provisioning procedures, see “Provisioning Domain Based Routing” section on page 4-73.

IP_ROUTE_SEL has the following data words:
- **inputDataType** (dw1)—Specifies the data that the Cisco PGW 2200 Softswitch uses to select the route. Valid values are:
  - 1 = Route selection against destination (user + host)
  - 2 = Route selection against destination host only
  - 3 = Route selection against source (user and host)
  - 4 = Route selection against source host only
  - 5 = Route selection against both destination (user and host) and source (user and host)
  - 6 = Route selection against both destination (host only) and source (host only)
  - 7 = Route selection against both destination (user and host) and source (host only) & source (user and host)
  - 8 = Route selection against both destination (host only) & source (user and host)
- **serviceName** (dw2)—Service name which must already exist in the service table (optional).
- **foundSetName** (dw3)—Result set name which must already exist in resultSet table, for execution conditional on a match being found in the table.
- **notFoundSetName** (dw4)—Result set name which must already exist in resultSet table, for execution conditional on no match being found in the table.

**IP_SET_SOURCE_DMN**
The IP_SET_SOURCE_DMN result type allows you to set the source domain name for domain-based calls. This result is supported for preanalysis and A and B number analysis only.

For more information on provisioning procedures, see “Provisioning Domain Based Routing” section on page 4-73.

IP_SET_SOURCE_DMN has the following data words:
- **dmnString** (dw1)—The name of the source domain.
- **applicationStatus** (dw2)—Specifies whether the command can override an existing domain name entry. The following values are valid:
  - 0 = The command can override an domain name entry.
  - 1 = The command cannot override an existing domain name entry.
- **applyTo** (dw3)—Specifies which source headers to which the PGW applies the command. The following values are valid:
  - 0 = Sets the PGW to apply the command to all source headers that are present.
  - 1 = Sets the PGW to apply the command to the current source header only.

**LOC_LABEL**
The LOC_LABEL result type is returned from A-number analysis (the calling number) or B-number analysis (the called number) and indicates the location label.

Dataword1 is the location label name, and can be as many as 20 alphanumeric characters.
MGCPDIALPKG

The analysis performed on a call is determined by the route the call takes to the Cisco PGW 2200 Softswitch. A call is considered either to be a TDM-switched call or an NAS call.

A call is considered to be a TDM-switched call if both call endpoints (that is, the originating and the terminating endpoints) are on the same gateway. As a result, hairpinning is required and no special result type from generic analysis is needed for this type of call.

However, for a NAS call, the MGCPDIALPKG result type is returned from generic analysis. As a result of this, the NAS package is used to set up the MGCP connection on gateways.

MGCPDIALPKG calls are based on the dial plan provisioning of the MGCPDIALPKG result type, which is provisioned against the B-number digit numbers. Thus MGCPDIALPKG calls take place on a call-by-call basis, which can occur along with regular voice calls, according to the B-number result type.

The Cisco PGW 2200 Softswitch receives an inbound call from the PSTN and enters B-number analysis (that is, the B-number). The MGCPDIALPKG result type is provisioned against the B-number. The following result types are available from generic analysis, which are based on the MGCPDIALPKG result type:

- Digital Data NAS Call
- Analog Data NAS Call
- Dynamic NAS Call

A dynamic call type is where the NAS advises the bearer type is digital only if transmission media requirements indicate 64 kbps unrestricted data service for the call. If the bearer type is not 64 kbps unrestricted data, the NAS is advised the call is analog and the NAS can determine, based on the bearer stream, the call type. This checking is made in generic analysis.

The MGCPDIALPKG result has two datawords: dataword1 and dataword2. Dataword1 has three different values (Digital, Analog, or Dynamic) that provide call type information.

If the result type from the B-number digit analysis is MGCPDIALPKG, and dataword1 is Digital or Analog, then conditional route analysis and Route analysis are not performed.

However, if the result type from the B-number digit analysis is MGCPDIALPKG, and dataword1 is Dynamic, then the bearer type is checked to see if it is 64 kbps unrestricted data. If the bearer type is 64 kbps unrestricted data, then the bearer type is set to DIGITAL. However, if the bearer type is not 64 kbps unrestricted data, then the bearer type is set to ANALOG.

No routing is performed if analysis receives an MGCPDIALPKG result type, since this is a data call to a one legged MGCP connection. The data call is connected to the 5350/5400/5800 gateway and therefore no circuit selection is needed.

With regard to dial plan data, the MGCPDIALPKG result is configured only when MGCPDIALPKG calls are required, and the result type is configured against the B-number in generic analysis only.

Dataword2 is a Boolean value (1 or 0) that indicates whether an ACM message is necessary in the call. Dataword2 is used to indicate whether to send (1) or not send (0) the ACM message.

When the MGCPDIALPKG result type is provisioned, it is provisioned in the dial plan only against the B-numbers and is read in generic analysis to determine if this call is an MGCP DIAL call.

For MML command configuration examples of intermediate MGCPDIALPKG results, see the “Adding the MGCPDIALPKG Result Type” section on page 4-17.
NEW_DIALPLAN

The NEW_DIALPLAN result type can be returned from Pre-analysis, A-number analysis, B-number analysis, or Cause analysis. It indicates the need to read the dial plan and check to see if a new dial plan should be used. If a new dial plan identity (CustGrpID) is found, this result initiates its selection. Once the new dial plan is selected, Pre-analysis can be restarted.

- **CustGrpID**—This dataword is relevant in all cases and supplies a customer group ID that is used to read the dial plan. It must be a valid customer group ID.

- **AnalysisType**—This dataword indicates the next stage of analysis, once the new dial plan is identified and invoked.

Valid values for dataword2 are dependent on the analysis stage from which the NEW_DIALPLAN result is returned, as shown in the following table.

If the NEW_DIALPLAN result is returned from Pre-analysis, only the following value is valid:
- 1 = Returns to the Pre-analysis stage in the new dial plan

If the NEW_DIALPLAN result is returned from A-number analysis, only the following value is valid:
- 0 = Default (dataword2 has no relevance from A-number analysis)

If the NEW_DIALPLAN result is returned from B-number analysis, the following values are valid:
- 1 = Returns to the Pre-analysis stage in the new dial plan
- 2 = Restart in B-number analysis in new dial plan

If the NEW_DIALPLAN result is returned from Cause analysis, only the following value is valid:
- 2 = Restart in B-number analysis in new dial plan

<table>
<thead>
<tr>
<th>Dataword2 (AnalysisType)</th>
<th>Dataword2 Value 1</th>
<th>Dataword2 Value 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>From Pre-analysis</td>
<td>Return to Pre-analysis in new dial plan</td>
<td>Not valid</td>
</tr>
<tr>
<td>From A-number analysis</td>
<td>Return to Pre-analysis in new dial plan</td>
<td>Not valid</td>
</tr>
<tr>
<td>From B-number Analysis</td>
<td>Return to Pre-analysis in new dial plan</td>
<td>Start B-number analysis in new dial plan</td>
</tr>
<tr>
<td>From Cause Analysis</td>
<td>Not valid</td>
<td>Start B-number analysis in new dial plan</td>
</tr>
</tbody>
</table>

The provisioning code checks to ensure that the new dial plan to be selected by the NEW_DIALPLAN result type is not the same as the current dial plan to avoid the possibility of a loop situation.

MGCP 1.0 on the PGW 2200 modifies the NEW_DIALPLAN result type to allow the PGW to re-start at the A Number stage of analysis. To use this setting, set dataword 2 to a value of 3.
Result Analysis

The Domain-Based Routing NEW_DIALPLAN has the following data words:

- custGrpId—Identifies the new dial plan to which the PGW switches.
- AnalysisType—Indicates the stage in which number analysis should start in the new dial plan. This data word applies to B-number analysis only. Valid values are as follows:
  - 0 = Default (dataword2 has no relevance from A-number analysis)
  - 1 = Start analysis at Pre-analysis stage
  - 2 = Start analysis at B-number stage
  - 3 = Start analysis at A-number stage (new value)

NUM_TRANS

The NUM_TRANS result type is returned from A-number (the calling number) or B-number analysis (the called number) indicating that one or more numbers encountered require full replacement.

This feature requires setting the *.FNTBehaviourOptions parameter in the XECfgParm.dat file on initial configuration. The *.FNTBehaviourOptions parameter has two valid values, 0 and 1. When *.FNTBehaviourOptions is enabled (set to value 1), if a successful number translation occurs, A/B/Redirecting number modifications through AMODDIG/BMODDIG/RMODDIG configured in the same result set with NUM_TRANS will get dropped.

If you are going to use this feature for the first time, you are recommended to set the value of *.FNTBehaviourOptions to 1. The value 0 is used for consistency with the existing behavior of the full number translations function.

See the “Provisioning Full Number Translations” section on page 4-69 for provisioning procedures of full number translations.

The NUM_TRANS result type has the following datawords:

- ServiceKey—An integer representing the previously provisioned Service Name in the Service table. This is a user-controlled key into the Times Ten query full number translation table. Digit strings stored in the full number translation table are case insensitive. That is to say, if digit strings that you provisioned contain alphabetic characters, the TimesTen database saves them as uppercase characters in the full number translation table.

  Note
  The service key must reference a previously provisioned service name.

- Number Type—An integer indicating the number type being translated. Valid values are:
  - 1 (CdPn)—Called party number
  - 2 (CgPn)—Calling party number
  - 3 (Rdn)—Redirecting number
  - 4 (Rdn and CgPn)—Calling party number and Redirecting number. Both numbers are replaced if the calling party number is found in the TimesTen database.
  - 5 (OCN)—Original called number.
- Nature of Address (NOA)—(Optional) An integer value that indicates the NOA value for the number type being translated. Valid values are 0 through 55.

  Note
  This field is updated only if a successful match is found in the full number translation table.
• Dial plan—(Optional) This is a 4-digit integer that represents the previously provisioned dial plan(s) in the Cisco PGW 2200 Softswitch. Valid values for this dataword are existing dial plan indexes, which are 0001 through 9999.

---

**Note**

The dial plan changes only if a successful lookup occurs in the full number translation table.

**Note**

The dial plan must reference a previously provisioned dial plan name.

When a successful NUM_TRANS lookup occurs, it takes precedence over all other results in the result set. If the NUM_TRANS result is not successful, all remaining results in the result set are performed. Thus it may be advisable to complete any dial plan changes before resuming number analysis. After a successful number replacement, the flexibility of this result can cause confusion in cases where A-number replacements are successful in B-number Analysis and B-number replacements are successful in A-number Analysis. In the dial plan, you can place A-number replacements in A-number analysis and B-number replacements in B-number analysis. Thus occurrences of replacements become more obvious and logical.

The following items further describe the behavior of the NUM_TRANS result type:

• NUM_TRANS result types can be present in both A-number analysis or B-number analysis.
• Because the NUM_TRANS result type causes an entire number replacement to occur, the nature of address may also be replaced.
• Both the NOA changes and dial plan changes provisioned against the NUM_TRANS result type are only acted on when a successful database lookup occurs.
• When a successful number translation occurs, a return to Pre-analysis is required.
• When a dial plan change is encountered, analysis begins at the Pre-analysis stage in the new dial plan.
• The NUM_TRANS result has priority in terms of the handling of all results and causes analysis to resume when a successful result is found.
• When multiple NUM_TRANS result types are encountered, longest matching is performed. As a result, the last successful database lookup against a specific number type is acted on, and any previous NUM_TRANS results against the same number are overwritten. As a result, a previous NUM_TRANS result may have successfully matched and a later NUM_TRANS result may fail; due of longest matching, only the last NUM_TRANS result encountered for the number type is effective.
• If a full number translation database lookup is not successful at any digit length, then any other digit modifications and result types are acted on.
• Although a NUM_TRANS result can be declared at any digit length, the number used for comparison purposes is the entire dialed number.
• For overlap sending, any NUM_TRANS result encountered causes a wait until all digits are received before a database comparison is performed.
• The number presented to the full number translation database is the full dialed number, without any other digit modifications that may have been encountered in other result types.
• If multiple NUM_TRANS result types, with different number types, are contained in a result set; but all NUM_TRANS result types indicate a dial plan change, then the longest match on the dial plan change occurs. Thus the dial plan change indicated in the last successful database lookup of a number type is used.
• A successful database lookup indicating a dial plan change overrides explicit dial plan change results that may also be present in a result set.

**ORIG_VPN_ID**
The ORIG_VPN_ID result type is returned from A-number analysis (the called number) indicating the originating VPN ID and if the originating index is on net or off net. Before you use this result type, you need to add the VPN ID by using `numan-add:customervpnid:custgrpid=<customer group ID>, name=<VPN ID>`. Then you can use an existing VPN ID for dataword1 of this result type.

This result type has the following datawords:

- **VPN ID** (dataword1)—Valid values are existing VPN IDs (8-digit alphanumeric character string).
- **VPN onnet profile index** (dataword2)—Valid values are a single integer from 1 to 8, with a default value of 5.
- **VPN offnet profile index** (dataword3)—Valid values are a single integer from 1 to 8, with a default value of 6.

**OTG_NUMBERING**
The OTG_NUMBERING result type returns information regarding the outgoing trunk group side (Terminating Call Control). This information sets the numbering criteria (that is, overlap or en bloc), and the minimum and maximum permitted digits for that side.

- **Numbering type**—0 = Closed numbering (en bloc), 1 = Open numbering (overlap).
- **Minimum and maximum digits**—This refers to the minimum number length and the maximum number length. (In the case of closed numbering, these values should be equal.)

**OVERRIDE_CALLIM**
The OVERRIDE_CALLIM result type indicates that the location label call overrides the call limiting value. Presence of the OVERRIDE_CALLIM result type indicates that for this call, any call limiting actions are ignored allowing it to being set up as soon as possible.

The OVERRIDE_CALLIM result type is available to Pre-analysis, A-number analysis, and B-number analysis. Since OVERRIDE_CALLIM is available to these analysis areas, the override indicator can be set for the following:

- **Calling Party Category** (CPC)—Pre-analysis
- **Calling party number Nature of Address** (NOA)—Pre-analysis
- **Called party number Nature of Address** (NOA)—Pre-analysis
- **Calling party number address digits**—A-number analysis
- **Called party number address digits**—B-number analysis

The OVERRIDE_CALLIM result type can be used for an emergency call or other high-priority calls. This result type allows those calls to be set up without any obstacles, such as call limiting. Even if LOC_LABEL results are collected, the presence of the OVERRIDE_CALLIM result type means that no call limiting actions are applied for this call.

**PERC_ROUTE**
The PERC_ROUTE result type provides an entry into the Percentage Routing lists. The Percentage Route list name is used as the starting point in the Routing analysis process.
PNMODDIG
The PNMODDIG result type modifies the presentation number received in any incoming message. This parameter populates or modifies a specified number of digits from any point in the GN-ACgPN, or Presentation Number.

Dataword1 (Application point) indicates the point (the digit) in the digit string that the Cisco PGW 2200 Softswitch begins applying the modification. The range is from 1 through the total number of digits in the digit string (32 maximum). Entering a value of “98” causes the removal of digits to begin at the end of the digit string and move backward to the beginning.

Dataword2 (Number of digits to remove) indicates the number of digits to remove. The range is from 0 through the number of digits remaining in the digit string from the application point (32 maximum). To remove all digits, regardless of the number of the number, enter the value 99.

Dataword3 (Modification name) indicates the name of the modification string. If required, this is a name that specifies the digit modification string that is to be inserted beginning at the application point.

PN_NPI_TYPE
The PN_NPI_TYPE result type is for NPI and PN. The call context is updated, including A-number screening indication, A-number presentation indication, A-number NPI value, generic number NOA value, generic number screening indication, generic number presentation indication, and CBI_IND for BTNUP and UKISUP protocol variants, based on generic analysis results.

All results are collected and then are processed in a logical order. First the Cisco PGW 2200 Softswitch checks for any call rejection cases (for example, Analysis failure, Cause, or Blacklist). Then, the Cisco PGW 2200 Softswitch handles any results that are processed before others (for example, screening (no point in additional processing if this does not pass)) or ported number handling where a number must be prefixed and then passed back in to start analysis again. Then any results, (for example, More information requests, and Test calls), and then finally all other results (ROUTE -Number modifications, and so on) are processed.

Dataword1 is the internal NPI value. The value range is 0 (default) through 10.

PN_NUMBER_TYPE
The PN_NUMBER_TYPE result type is used to modify the number type of the presentation number. The NOA modification field of the presentation number or the generic number is modified.

Dataword1 value is the internal NOA value. The value range is 0 (default) through 53.

PN_PRES_IND
The PN_PRES_IND result type is the presentation indicator of the presentation number, or the generic number is modified with this result type.

Dataword1 is the presentation number indicator value. The value range is 1 through 3.

  * 1 = Restricted
  * 2 = Allowed
  * 3 = Unavailable
PN_SCREEN_IND
The PN_SCREEN_IND result type is the screening indicator of the presentation number, or the generic number is modified with this result type.

Dataword1 is the presentation number screening indicator value. The value range is 1 through 5.
- 1 = NP (Network Provided)
- 2 = UPVP (user provided verified and passed)
- 3 = UPNV (user provided not verified)
- 4 = UPVF (user provided verified and failed)
- 5 = spare

PREFIX_CONVERT
The PREFIX_CONVERT result type allows the Cisco PGW 2200 Softswitch to support prefix modification for connected number, redirection number, and transferred number. PREFIX_CONVERT can work for SIP-to-ISUP, ISUP-to-SIP, and ISUP-to-ISUP connected numbers. It cannot work for SIP-to-SIP connected number.

Note  The prefix modifications are based on the original calling/called/generic number received on the originating side.

REDIRECT
The REDIRECT result type allows a call to be redirected based on call properties such as the A number or B number. REDIRECT can be provisioned for A or B Number analysis.

ServiceKey—Dataword1 (dw1) is an integer representation of the name of the provisioned service (ServiceName).

Note  The redirect server feature is enabled for DPNSS only. It does not work for SIP.

RETRY_ACTION
The RETRY_ACTION result type can be provisioned only in Cause analysis and provides the required actions with regard to route advance, reattempt, or redirection. This result has one integer data word that represents the required action. You can also configure the stage of analysis in which the Cisco PGW 2200 Softswitch restarts when retrying a call. This capability provides consistent redirection handling for E.164 and non-E.164 calls.

RETRY_ACTION 1 has the following datawords:
- RetryType (dw1)—Manner in which the Cisco PGW 2200 Softswitch retries the call. Valid values:
  - 1 = Reattempt
  - 2 = TGAdvance
  - 3 = Redirect

Reattempt: The reattempt function is controlled by the “Reattempts” value that is provisioned in Trunk Group Data. Reattempts only take place up to the limit of this provisioned value. If the counter is exceeded, then instead of a Reattempt a trunk group advance takes place.

TGAdvance: A property “MaxNumTGAdvances” contains a value defined in the XECfgParm.dat file. Should the value limit be met or exceeded, the call is released using the existing cause (Treated Cause result). Result-type is not fully supported in a SIP and SS7 protocol interworking scenario.
Redirect: Redirection processing is only for the ISUP protocols and is limited to a maximum of 5 redirections by the system property “RedirMax”. The MGC checks within Generic Analysis before processing the result, ensuring it is processed only if the value is less than 5 and less than the setting of the RedirMax property. Should the received counter value already be at 5, or exceed the configured threshold the result is ignored and the call released by use of the existing cause value. The Generic Analysis module returns a “Treated Cause” result. Clear the call by normal release mechanisms, then call a common routine that makes a new analysis request for A, B, and Routing analysis (forwarding the Re-direction number as the B-number). The expected response is a new trunk group upon which to attempt circuit selection.

- redirAnPhase (dw2)—Phase of analysis in which the Cisco PGW 2200 Softswitch restarts when retrying a call. Valid values:
  - 0 = Redirection next analysis phase B-number analysis (default value)
  - 1 = Redirection next analysis phase Pre-Analysis

RMODDIG

The RMODDIG result type is for digit modification on the redirecting number. The capability exists to remove a specified number of digits from any point in the redirecting digit string and replace them with whatever digits are required.

The RMODDIG result type has the following datawords:

- **Application point**—The point (digit) in the digit string to begin applying the modification.
  The range is from 1 through the total number of digits in the digit string (32 maximum). Entering a value of “98” causes the removal of digits to begin at the end of the digit string and move backward.

- **Number of digits to remove**—The range is from 0 through the number of digits remaining in the digit string from the application point (32 maximum). To remove the entire number, regardless of the number of digits it contains, enter the value “99” for this dataword.

- **Modification name**—If required, this is a name that specifies the digit modification string that is to be inserted beginning at the application point.

- **Remove Leading Digits**—When dw4 is set to 0, the Cisco PGW 2200 Softswitch uses the RMODDIG result type as normal. When dw4 is set to 1, the Cisco PGW 2200 Softswitch removes leading digits from the Redirecting Number and the original called number. When dw4 is set to 2, if the incoming redirecting number is NULL, the Cisco PGW 2200 Softswitch does not insert one redirecting number. For other cases, the Cisco PGW 2200 Softswitch behaves as before.

**Note**  
If the leading digit of the original called number is 0, it can be removed as the Redirecting Number when dw4 is set to 1.

Dataword rules:

- Dataword1 must be 1 through 32 or 98.
- Dataword2 must be 0 through 32 or 99.
- Dataword3 must be 0 or an existing digit modification name.
- Dataword4 must be 0 through 2.
For example, if the application point = 1, the number of digits to remove = 5, and the modification name gives a result of 1321, then begin at the start of the digit string, remove 5 digits, and replace them with the digit string 1321. This yields a redirecting number as follows:

- Redirecting number received pre-analysis = 01444 567891
- Redirecting number post analysis = 1321 567891

For example, if the application point = 98, the number of digits to remove = 4, and the modification name gives a result of 1321, then begin at the end of the digit string, remove 4 digits, and replace them with the digit string 1321. This yields a redirecting number as follows:

- Redirecting number received pre-analysis = 12345567891
- Redirecting number post analysis = 12345561321

Depending on the analysis area that invokes it, the RMODDIG result type has different functions. The following are examples of these different functions:

- In Pre-Analysis there are currently four serial stages that can produce the RMODDIG result type. In Pre-analysis, the results are cumulative. For example, if the CPC stage generates an RMODDIG result type, then the redirecting number is modified according to the result and this modified number then is the new redirecting-number passed as input to the next Pre-analysis stage (TMR analysis). If the TMR analysis provokes another RMODDIG result type, then it further modifies the number and so on. Even though multiple modifications like this would seem excessive and unnecessary, the capability exists to ensure the required flexibility is provided.

- In Number analysis (A-number or B-number), functionality is different. Here digit analysis is applied (digit by digit) and it is possible to have the RMODDIG result type at multiple points if required. However, it is only the last modification result type that is applied.

**Note** Digit modification is applied to the initial number input to this analysis stage. There is no cumulative digit modification performed.

For example, if the received redirecting number is 1234 and at “1” an RMODDIG result type is received making the number 441234, the digit string is modified and analysis continues according to the digit analysis configuration. If another RMODDIG result type is received at 1234, making the number 551234, the earlier RMODDIG result type (“1”) is discarded and the number now sent forward is 551234.

**R_NUMBER_TYPE**

The R_NUMBER_TYPE result type lets you change the redirecting number type nature of address (NOA) from that presented in the IAM or Setup message. This result type is available to Pre-analysis, A-number analysis, B-number analysis, Cause analysis. R_NUMBER_TYPE uses the following data words:

- Dataword1 (dw1) provides the Cisco PGW 2200 Softswitch internal call context value for the (NOA) of the redirecting number.
- Dataword2 (dw2) determines whether the Cisco PGW 2200 Softswitch updates the nature of address (NOA) of the original called number (OCN). Dataword2 has the following values:
  - 0 — The NOA of the OCN is not modified. This is the default value.
  - 1 — The NOA of the OCN is changed according to the redirecting number. For example, if dw1 is set to 5 and dw2 is set to 1, the NOA of the redirecting number and the NOA of the OCN are changed to “international.”
**Note**
The NOA value needs to be the MGC internal value and not the protocol-specific value. See Appendix A, “NOA and NPI Codes, CPC and TMR Values” for specific protocol values.

**ROUTE**
The ROUTE result type supplies a Route List name, which is used as a starting point in the Routing analysis process.

**Note**
The ROUTE result type is not used in a Cisco PGW 2200 Softswitch (signaling) application.

**ROUTE_PREFERENCE**
The ROUTE_PREFERENCE result type is applicable only to A-number analysis. It provides an indication of preferred egress trunk group type in relation to the received A-number. The value set in the result (see the following list of possible values) is used within the Routing analysis stage and provides a further bias to the trunk group selection algorithms.

The possible values for ROUTE_PREFERENCE are as follows:

- 0 = RTE_SEL_DONT_CARE
- 1 = RTE_SEL_ATM_ESSENTIAL
- 2 = RTE_SEL_ATM_PREFERRED
- 3 = RTE_SEL_ATM_EXCLUDED
- 4 = RTE_SEL_IP_ESSENTIAL
- 5 = RTE_SEL_IP_PREFERRED
- 6 = RTE_SEL_IP_EXCLUDED
- 7 = RTE_SEL_TDM_ESSENTIAL
- 8 = RTE_SEL_TDM_PREFERRED
- 9 = RTE_SEL_TDM_EXCLUDED

**RTRN_START_ANAL**
The RTRN_START_ANAL result type performs different actions depending on what stage of the analysis generates it:

- In B-number analysis, this result type causes the carrier code prefix, if any, to be deleted and B-number analysis is restarted with the modified B-number.
- In Cause analysis, this result type initiates a return to B-number analysis; however, the B-number to be analyzed will include any modifications and any NOA call type modifications.
SCREENING

The SCREENING result type delivered from either A-number or B-number analysis indicates that the A-number or redirecting number must be screened against the screening files configured for a specific customer group ID. ScreenType values 3 and 4 are added in MGC software Release 9.4(1), allowing screening to occur using the global customer group ID GLBL as the input key. Dataword1 (screen type) identifies the type of screening that must be requested. Dataword2 (service name) is only used when screening is requested from B-number analysis and identifies the database list of A-Numbers and redirecting numbers that must be screened, which are appropriate to the B-number. Dataword3 is the index to the dialPlan selection table if the screening passes. Dataword4 is the index to the dialPlan selection table if the screening fails.

- **ScreenType**—Must be one of the following:
  - **1 = Whitelist**—If the presented A-number or redirecting number is not found in the screening files, then the screening is considered to have failed and the call is released.
  - **2 = Blacklist**—If the presented A-number or redirecting number is found in the screening files, then the screening is considered to have failed and the call is released.
  - **3 = Global Whitelist**—If the presented A-number or redirecting number is not found in the screening files, then the screening is considered to have failed and the call is released. Added in software Release 9.4(1).
  - **4 = Global Blacklist**—If the presented A-number or redirecting number is found in the screening files, then the screening is considered to have failed and the call is released. Added in software Release 9.4(1).

- **Service Name**—When screening is triggered by B-number analysis, a service name (such as “800,” “900,” or “FreePhone”) is used to identify which list of calling numbers (A-numbers) is associated with that service. The service name is passed, as read, when the screening request is made.

  **Note** Service names are limited to 10 alphanumeric characters. Spaces are not allowed.

- **Pass_DpIdx**—(optional) Provides an index for dial plan selection if the screening type, which is available only for A-number analysis, passes. Also includes B-number analysis in software Release 9.6(1). If the screening passes, the dial plan index from this dataword is used to cause a dial plan change and then processing returns to pre-analysis. If no index value is present, number analysis continues. Added in software Release 9.4(1).

- **Fail_DpIdx**—(optional) Provides an index for dial plan selection if the screening type, which is available only for A-number analysis, fails. Also includes B-number analysis in software Release 9.6(1). If the screening fails, the dial plan index from this dataword is used to cause a dial plan change, and then processing returns to pre-analysis. If no index value is present, number analysis continues. Added in software Release 9.4(1).

SCRIPT

The SCRIPT result type can be provisioned for B-number analysis, and is an end-of-analysis result type.

- **ScriptId**—Dataword1 (dw1) is an integer and provides an index into the Script table in the database, where the details (for example, Gateway type, script type, script location, and optional script parameters) are stored.

- **CallType**—Dataword2 (dw2) is an integer and indicates the CallType associated with this result type. A value of 1 hands over call control to the gateway, with script invocation. Currently, only this result type is supported.
• **AcmReqdInd**—Dataword3 (dw3) is an integer and indicates whether an optional ACM is to be sent when the confirmation of script invocation is received (for dw2, CallType = 1).

• Dataword4 (dw4) is not used.

**SIPI_CONTROL**
The SIPI_CONTROL result type allows you to enable the SIP-I route preference and to overwrite the outgoing SIP-I related configuration parameters on the outgoing trunk group. Currently this result type is used to enable the SIP-I route preference only.

Dataword1 enables the route preference. A value of 1 enables the route preference.

**SIPTNS**
The Carrier Identification Code is a three- or four-digit code used in routing tables to identify the network that serves a remote user when a call is routed over many different networks. The SIP CIC parameter transmits the CIC value from the SIP network to the ISDN. The SIP CIC parameter is carried in SIP INVITE requests and maps to the ISDN Transit Network Selection Information Element (TNS IE).

The SIP TNS result type allows you to map the CIC from the SIP INVITE parameter to the TNS IE in the outgoing IAM message for ANSI ISUP. The Cisco PGW 2200 Softswitch uses the called party number (B number) and the SIP CIC to populate the TNS IE.

Table 1-6 shows the format of the TNS parameter with a 4-digit carrier identification code.

<table>
<thead>
<tr>
<th>Table 1-6 Transit Network Selection</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
</tr>
<tr>
<td>H</td>
</tr>
<tr>
<td>Spare</td>
</tr>
<tr>
<td>Digit 2</td>
</tr>
<tr>
<td>Digit 4</td>
</tr>
<tr>
<td>Circuit code</td>
</tr>
</tbody>
</table>

The Cisco PGW 2200 Softswitch uses the following values to populate the TNS:

• Type of network identification—010 (National Network Identification)

• Network identification plan—0010

• Digits 1–4—The Cisco PGW 2200 Softswitch inserts the four-digit CIC value from the SIP URI.

• Circuit code—The Cisco PGW 2200 Softswitch uses a binary version of the first digit of the called party number (B number).

**Note** You can also manually set the circuit code value using the circuit code data word.

The SIPTNS result type has the following data words:

• Circuit code value: Sets the ISDN circuit identification code value. This value is the decimal version of a binary number; for example, the value 3 sets the circuit code value to 0011. This value overrides the circuit code value derived from the called party number (B number).

Valid values: 0–15
TERM_INFO
The TERM_INFO result type is used to enable full called number analysis to make a call to the Number Termination table (TERMTBL), which provides a route list name to start routing analysis. You can reduce the size of a dial plan to achieve a full called number analysis. The TERM_INFO result type is configured early in the B-number analysis. All actions are implicit by the presence of this result type; consequently, there are no datawords accompanying this result.

TESTCALLDETECTED
The TESTCALLDETECTED result type is used to indicate that the called number (B-number) is associated with a test call. The parameters associated with this result type are:

- **Test Line Type**—Can be one of the following values:
  - 0 = Quiet termination (qt)—this is the default value
  - 1 = Old milliwatt (1000 Hz)
  - 2 = New milliwatt (1004 Hz)
  - 3 = Really new milliwatt (1013.8 Hz)
  - 4 = Tone off
- **Test Line Duration**—The duration of the test signal in milliseconds. The range of duration values is 0 through 65,535 milliseconds. The default value is 0.
- **Test Line Name**—Can be up to 20 alphanumeric characters. The test line name is always converted to lowercase in the provisioning object library.

VIDEO_ALLOWED
The VIDEO_ALLOWED result type enables the Cisco PGW 2200 Softswitch to allow or prohibit video calls at the dial plan level.

There are two levels of video call admission control, the dial plan level and the trunk group level. If video calls are allowed at the trunk group level but prohibited at the dial plan level, video calls are prohibited. If video calls are prohibited at the trunk group level but allowed at the dial plan level, video calls are allowed.

This result type provides you the flexibility to include video call admission control in the number analysis. For example, you can prohibit video calls whose B-numbers start with 909.

Dataword1 specifies whether the Cisco PGW 2200 Softswitch allows or prohibits video calls:

- 0 = Prohibits video calls at the dial plan level.
- 1 = Allows video calls at the dial plan level.

If you do not provision the VIDEO_ALLOWED result type, the Cisco PGW 2200 Softswitch allows video calls at the dial plan level by default.

WHITELIST
The WHITELIST result type returned from B-number analysis indicates that the called number is valid and that call processing can proceed. No datawords are used and any call processing action is implicit by the presence of the result type. No call screening is associated with this result type.
When you are using a default result type on a Cisco PGW 2200 Softswitch, in signaling mode, results of the WHITELIST result type for B-numbers should contain routing information to prevent the analysis from dropping through to the default result type. This result takes the place of the ROUTE result used with the Cisco PGW 2200 Softswitch in call control mode, and ensures that the call completes. Absence of the WHITELIST result type invokes the default result type on a Cisco PGW 2200 Softswitch in signaling mode.

Processing Multiple Result Types

When Pre-analysis, A-number analysis, and B-number analysis, are performed the results are collected, and then processed in a logical sequence. This sequence ensures that no further processing is carried out if there has been analysis failure, or if there is a cause or blacklist result, in which case the call clear down can be provoked.

Additionally, there is a separation of result handling that allows screening to take place before the Cisco PGW 2200 Softswitch performs other result types, such as number modifications or route results. Another scenario is that of Local Number Portability (LNP) handling by the on-board database that is carried out at an early stage of analysis to ensure that once the B-number is modified by adding or removing a prefix code, B-number analysis can be re-started (finally leading to a Routing result).

All result matches for a digit string are added together and only duplicate result types are overwritten by the longest match.

As a result, there are a number of transparent stages in result processing, which may not be apparent when provisioning, at such time it may appear that the defined order has a bearing on the final result. It is also important to note that where the same digit analysis root is used as a fork for several different result set actions, some result types (even though defined within different result sets) may impact one another.

This means that at certain points you can encounter results that stop processing at a point and must make an immediate response for action, for example, whether to provoke call clear down or a more information backward request (as shown in Figure 1-7).
The tables in the following sections categorize the result types so you can understand at what point in the logical hierarchy they are processed. The results are separately tabulated for A-number, B-number, and Cause analysis stages.

### Pre-analysis Stages

Table 1-7 lists the result types for Pre-analysis. Result types include direct response and all others.

**Table 1-7 Pre-analysis**

<table>
<thead>
<tr>
<th>Direct Response Results</th>
<th>All Other Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internal Analysis Failure, BLACKLIST</td>
<td>AMODDIG,</td>
</tr>
<tr>
<td></td>
<td>A_NUMBER_TYPE,</td>
</tr>
<tr>
<td></td>
<td>BMODDIG,</td>
</tr>
<tr>
<td></td>
<td>B_NUMBER_TYPE,</td>
</tr>
<tr>
<td></td>
<td>CALL_CUTOFF_TIMER,</td>
</tr>
<tr>
<td></td>
<td>COND_ROUTE,</td>
</tr>
<tr>
<td></td>
<td>INC_NUMBERING,</td>
</tr>
<tr>
<td></td>
<td>NEW_DIALPLAN,</td>
</tr>
<tr>
<td></td>
<td>PERC_ROUTE,</td>
</tr>
<tr>
<td></td>
<td>RMODDIG</td>
</tr>
<tr>
<td></td>
<td>R_NUMBER_TYPE</td>
</tr>
<tr>
<td></td>
<td>ROUTE</td>
</tr>
</tbody>
</table>
A-number Analysis Stages

Table 1-8 lists the result types for A-number analysis. Result types include direct response, early processed, remaining direct responses, and all others.

Table 1-8  A-number Analysis

<table>
<thead>
<tr>
<th>Direct Response Results</th>
<th>Early Processed Results</th>
<th>Remaining Direct Response Results</th>
<th>All Other Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internal Analysis</td>
<td>SCREENING</td>
<td>CHARGEORIGIN, CPCMOD</td>
<td>AMODDIG, A_NUMBER_TYPE, A_NUM_DP_TABLE, BMODDIG, B_NUMBER_TYPE, CALL_CUTOFF_TIMER, CG_PRES_IND, NEW_DIALPLAN, RMODDIG R_NUMBER_TYPE ROUTE_PREFERENCE</td>
</tr>
<tr>
<td>Failure, BLACKLIST, CAUSE, CLI_NBR_LENGTH</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

B-number Analysis Stages

Table 1-9 lists the result types for B-number analysis. Result types include direct response, early processed, remaining direct responses, and all others.
Table 1-9  B-number Analysis

<table>
<thead>
<tr>
<th>Direct Response Results</th>
<th>Early Processed Results</th>
<th>Remaining Direct Response Results</th>
<th>All Other Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internal Analysis Failure, BLACKLIST, CAUSE</td>
<td>E_PORTED_NUM, E_ROUTE_NUM, SCREENING</td>
<td>BSM_REQ, CPC_REQ, CLI_REQ, DIGIT_REQ, FSM_REQ, RTN_START_ANAL, TESTCALLDETECTED</td>
<td>ADDRESSCLASS, AMODDIG, ANNOUNCEMENT, A_NUMBER_TYPE, A_NUM_DP_TABLE, BMODDIG, B_NUMBER_TYPE, CALL_CUTOFF TIMER, CG_PRES_IND, CHARGE, CODEC, COND_ROUTE, DATA.Exchange, INC_NUMBERING, IN_TRIGGER, MGCPDIALPKG, NEW_DIALPLAN, OTG_NUMBERING, PERC_ROUTE, RMODDIG R_NUMBER_TYPE ROUTE, TERM_INFO, WHITELIST</td>
</tr>
</tbody>
</table>

Cause Analysis Stages

Table 1-10 lists the result types for Cause analysis. Result types include direct response results and all other results.

Table 1-10  Cause Analysis

<table>
<thead>
<tr>
<th>Direct Response Results</th>
<th>Early Processed Results</th>
<th>Remaining Direct Response Results</th>
<th>All Other Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internal Analysis Failure, CAUSE, RETRY_ACTION</td>
<td></td>
<td></td>
<td>ANNOUNCEMENT, BMODDIG, B_NUMBER_TYPE, COND_ROUTE, NEW_DIALPLAN, PERC_ROUTE, RMODDIG R_NUMBER_TYPE ROUTE, RTN_START_B_AN</td>
</tr>
</tbody>
</table>
The following examples provide the call processing order for different combinations of results in a result set:

**Example 1**
The result set includes ROUTE, BMODDIG, B_NUMBER_TYPE, SCREENING

The actual processing sequence is:
- SCREENING,
- BMODDIG,
- B_NUMBER_TYPE,
- ROUTE

**Example 2**
The result set includes E_PORTED_NUM, SCREENING, and ROUTE.

The actual processing sequence is:
- SCREENING,
- E_PORTED_NUM—prefix B-number with routing number and re-start B-analysis,
- ROUTE (from new B-number analysis)

**Example 3**
The result set includes MORE_INFO (BSM_REQ, CPC_REQ, CLI_REQ, DIGIT_REQ, and FSM_REQ), ROUTE, and BMODDIG.

The actual processing sequence is:
- MORE_INFO—response to the call control module and provokes backward request for CLI from OCC protocol. OCC protocol responds with CLI—new analysis request, (MORE_INFO)—now ignored since CLI is present,
- ROUTE,
- BMODDIG

**Example 4**
The result set includes NEW_DIALPLAN, SCREENING, and BMODDIG.

The actual processing sequence is:
- SCREENING,
- BMODDIG,
- NEW_DIALPLAN

**Handling Multiple Occurrences of Result Types**

**Pre-analysis Processing of Result Types**

Multiple repeated result types occurring in Pre-analysis use the following rules:

ROUTE, COND_ROUTE, PERC_ROUTE, INC_NUMBERING, CALL_CUTOFF_TIMER, NEW_DIALPLAN, and R_NUMBER_TYPE Result Types

Each result overwrites the previous result; the last result is the one acted on.
AMODDIG, BMODDIG, A_NUMBER_TYPE, B_NUMBER_TYPE, and RMODDIG Result Types
Cumulative through Pre-analysis, if one stage modifies a number this number becomes the input to the next stage (and is the basis for any further modifications).

BLACKLIST Result Type
Exits from current Pre-analysis stage and clears the call.

A-number and B-number Analysis Processing of Result Types

ROUTE, COND_ROUTE, and PERC_ROUTE Result Types
The last result (of any one type) retrieved deletes and overwrites all previous result types of the same result.

IN_TRIGGER Result Type
When you are provisioning IN_TRIGGER result types, it is helpful to also provision a ROUTE result within the same result set.

On calls with an IN_TRIGGER result, a call is made to the SCP for number translation. If the number is recognized as a ported number, a translated “routing number” (LRN) is returned from the SCP and the MGC analyses and routes according to this number (carrying the original called number within the GAP parameter). If the SCP does not recognize the number and has no translation number to offer, it may return the same called number. If this happens, analysis is the same—delivering the IN_TRIGGER result. This could be a potential problem that results in looping between the MGC and the SCP.

To prevent this looping, protection has been designed in to the MGC to guard against looping. Thus, if a default ROUTE result is provisioned with the IN_TRIGGER result, the route is picked up and used to route the call.

TERM_INFO Result Type
When provisioning a TERM_INFO result, you may also be required to configure a default ROUTE result (in the same result set) for use if the call cannot be routed using TERM_INFO. Within generic analysis, the TERM_INFO is always handled first. So there is no problem to put a ROUTE result at the same stage of digit analysis, or at any other point in the decoding process.

The following actions can occur where no default ROUTE result is available:

If no default ROUTE result has been provisioned in the result set and an error occurs during analysis causing no ROUTE result to be retrieved or as described previously, a double IN_TRIGGER result occurred, then analysis handling is dependent on the MGC configuration.

In a call control configuration, this is fatal, since the call cannot be set up, causing the result “Analysis failure” to be returned with a cause set to “Temporary failure”.

In a signaling configuration, a result of “Analysis Performed” is returned to the call control module. A determination is then made as to the relevant actions. If the minimum digits on the OCC side are met or exceeded, the call is continued. If the number of digits is not met, the cause “Address incomplete” is set and Cause analysis is invoked pending release of the call.

B-number Analysis NEW_DIALPLAN Result Type
If a NEW_DIALPLAN result is retrieved from B-number analysis once the dialplan is changed over, call processing can re-start at either the Pre-analysis or B-number analysis stages, which is an option configured in the NEW_DIALPLAN result type data.
ROUTE, COND_ROUTE, and PERC_ROUTE Result Types

When processing, any duplicate results of these types will overwrite any previous one. The one taken forward to start Routing analysis will then be the last result retrieved.

In Enbloc working this is clear as all digits are received at the outset, analyzed one by one and number analysis is therefore completed, allowing Routing analysis to begin. In Overlap working however, there will be no Routing Analysis until Number-analysis is complete. This means that even if duplicate ROUTE, COND_ROUTE, and PERC_ROUTE results are provisioned they are ignored until “Analysis-complete” is detected. At that point, the last one retrieved is the one used to provoke Routing analysis actions.

This enables a default capability in digit analysis such that an early “default” routing result can be provisioned for routing (for example, to an operator center) that is overwritten if correct routing according to a full area code decode takes place.

ANNOUNCEMENT Result Type

When handling this result, any duplicate result of this type overwrites the previous one. The ANNOUNCEMENT result returned to the call control module for action is the last retrieved. There is also a B-number modification associated with this result type that is always applied to the original B-number as received from the call control module.

AMODDIG, BMODDIG, and RMODDIG Result Types

Once all the Pre-analysis stages are complete the final modified numbers become input to the A/B-number analysis stage.

During A/B-number analysis, multiple number modification results of the same type cause the number in question to be modified but only by the last result of this type retrieved. Any previous modification is deleted and the new modification is applied to the original received A/B-number, not to the previously modified A/B-number.

It should also be noted that if A-number analysis modifies the A/B-number then this is the number that becomes input to the B-number analysis stage.

The number returned to the call control module for action is the last modified number. This avoids multiple modifications to the same number, which could result in an erroneous or corrupted number.

CPC_REQ, CLI_REQ, BSM_REQ, and FSM_REQ Result Types

When handling the above results, any duplicate result of these types will overwrite the previous one of the same type. The one returned to the call control module for action will be the last retrieved.

Before any action, a check is made to see if the requested data is already stored in Call Context. If it is, the result is ignored.

IN_TRIGGER Result Type

When handling this result, any duplicate result of this type will overwrite the previous one. The last result returned to the call control module for action will be the last result retrieved.

Initial checks are made with this result type to ensure that the call is not an Operator destined call, not 950-xxxxx, and not a CarrierID routed call (TNS present). In these cases a call to the SCP is not a valid action, so the result type would be ignored.
DATA_EXCHANGE Result Type
When handling, any duplicate result of this type will overwrite the previous one.
This result actions a “swap” of data from one Call Context location to another. That swap work is handled in the subsequent result-handling routine. That is to say, when the results are retrieved from the provisioned values, they simply overwrite each other. The final result is passed to the handling routine. If a data exchange is performed, (today only for LNP call processing), it is flagged, and if after the SCP call new analysis erroneously returns this result type again it will be ignored.

E_PORTED_NUM, E_ROUTE_NUM, and TERM_INFO Result Types
It is not expected that there would ever be a need to provision these results twice and this would add no value as the result merely indicates a read of the Ported list/Number Termination list. Thus last retrieved result causes its corresponding action to be taken.

RTN_START_ANAL Result Type
It would not be logical to have a repeated version of this result type, but if it occurs the last retrieved result will provoke the required action (in this stage of analysis this simply means re-starting B-number analysis again with an optionally modified b-number). This result carries no data that would be overwritten.

WHITELIST Result Type
Only available in B-number analysis, multiple results of this type are not expected and would not change result handling once one had been found. The bottom line is that the last retrieved result causes its corresponding action to be taken.

Result Types Appropriate to Default Routing Use
In the A- and B-number analysis areas, you may want to provision the dialplan so there are repeats (multiple occurrences) of certain result types. This allows some additional capabilities that can assist you in providing some network requirements.
With some results, it can help to have a “default” routing result that is selected at an early stage of digit analysis. This default routing result can be overwritten with another route result later in the decoding process. Thus, if for instance some provisioning has been overlooked, the call still routes according to the early default result. An example of this would be to set a ROUTE result to an operator center at the decode of 703, while the decode of 703484 routes to the Herndon area.

Note
The ROUTE result handling is separate to the functionality implemented for default handling.

When provisioned, the following result types can be augmented with a default routing result (that is, multiple provisioned ROUTE results). For each result type, an explanation is provided outlining why this is appropriate.

Cause Analysis Processing
In Cause analysis, some results can return specific data to immediately provoke tear down of the call. However, for others (for example, ROUTE or COND_ROUTE) it is necessary to return the result to routing and allow it to first clear and delete the existing terminating side, then re-invoke analysis with the stored results. This ultimately allows a new terminating side to be set up and the call re-routed forward.
Cause Analysis Handling for NEW_DIALPLAN Result Type
If a NEW_DIALPLAN result is retrieved from Cause analysis once the dialplan is changed over, call processing re-starts at the B-number analysis stage, which is the only option available from Cause analysis.

Note
A maximum of three dial plan changes, after the initial dial plan, is permitted.

Cause Analysis Handling for Multiple Result Type Occurrences

ROUTE, COND_ROUTE, and PERC_ROUTE Result Types
With these three result types, the last result retrieved overwrites any previous result and the last result becomes the one that is returned to analysis for action.

BMODDIG, B_NUMBER_TYPE, CAUSE, NEW_DIALPLAN, and RMODDIG Result Types
With these results from the Cause analysis area, any duplicated result types overwrite the previous one, and the last retrieved result will be the one used to provoke the required action.

RETRY_ACTION Result Type
It would not be logical to have a repeated version of this result type, but if it occurs the last retrieved will overwrite any previous result and will dictate the required actions.

RTN_START_ANAL Result Type
It would not be logical to have a repeated version of this result type, but if it occurs the last retrieved will provoke the required action (from this analysis stage this means clear the existing TCC side and then re-call analysis requesting B-number and routing analysis). This result carries no data that is overwritten.

ANNOUNCEMENT Result Type
When handling, any duplicate result of this type, the following result overwrites the previous result. The result returned to be performed is the last result retrieved. There is also a B-number modification associated with this result type that is applied to the original B-number as received.

Mixed Final Result Handling

If more than one final result is retrieved the following sections describe how call processing is performed.

Routing Result Types with an ANNOUNCEMENT Result Type
If either a ROUTE, COND_ROUTE, or PERC_ROUTE result and an ANNOUNCEMENT result are retrieved in a completed analysis flow (for example, from one result set), only the last retrieved result type is performed. If these two results are received together, they are mutually exclusive, and only the last encountered result type is performed.

Example 1: If the called number 867-1234 is provisioned with a ROUTE result at 867 and an announcement at 867-123, the ANNOUNCEMENT result is the one returned for action.

Example 2: Conversely if the called number 867-1234 is provisioned with an ANNOUNCEMENT result at 867 and a route at 867-123, the ROUTE result is the one returned for action.

A default Route or Announcement could still be set (as described in the previous section), or even a default for both, but ultimately if both result types are present and analysis is complete, the rule of the last retrieved result type being returned applies.
Routing Result Types with a TERM_INFO Result Type

If a combination of ROUTE or COND_ROUTE or PERC_ROUTE result with a TERM_INFO result occurs in a completed analysis flow, the order of processing is that the TERM_INFO result is always applied first. If this achieves successful final routing the other routing result is deleted and the call is progressed. If the TERM_INFO result is not successful in determining final routing, then the ROUTING result is used.

The same applies if an ANNOUNCEMENT result and TERM_INFO result combination occurs.

It is still possible to have successive ANNOUNCEMENT or ROUTE results with a TERM_INFO result, but the action described before in this section applies to the ANNOUNCEMENT/ROUTE results, and TERM_INFO is processed first.

Processing Dial Plan Longest Match

This feature (introduced in Release 9.6) provides support for using the longest match in a dial plan even when a new dial plan matches a shorter digit string. Formerly, with various result types, like ROUTE, CAUSE, ANNOUNCEMENT, the dial plan changeover was forced, and so the longest match was ignored.

With the introduction of the new Dial Plan Longest Match feature, the Cisco PGW 2200 Softswitch uses the longest dial plan match to select the best result type. Consequently, it will not jump to a new dial plan if there is another terminal result that has a potentially longer match. This applies to all of the results mentioned for A-analysis and B-analysis in the “Longest Match in A-Number Analysis” section on page 1-67 and “Longest Match in B-Number Analysis” section on page 1-67.

The Dial Plan Longest Match feature is further explained in the following two sections:

- **Basic Result Analysis**—explains the current call processing capability
- **New Call Processing Behavior**—explains the new functionality with the introduction of the Dial Plan Longest Match feature.

Basic Result Analysis

This section explains the basic result analysis based on the previous call processing capability.

Result analysis enables you to group actions into result sets that can be attached at different points of analysis. The main attachment points are pre-analysis, A-number analysis, B-number analysis, and cause analysis.

When you are configuring results, certain result types require extra configuration to provide additional data. The following are examples of two such result types.

- **Number modification**, in which the digits are inserted into a number. These new digits must be configured first and stored before the actual result, which will make use of these digits, is defined. For example, if the B-number is 4841234 and the intention with a B-number modification (BMODDIG result) is to insert 703 at the front of the number, the “703” digit string must be created first. Once the digit string is created, the actual B-number modification result can be defined through use of the "703" digit string data.

- **When A-number screening is required**, if the screening is triggered from the B-number digit analysis, it is necessary to identify the database area that contains the A-number screening data for calls destined to this particular B-number. The database area is called the Service name. The service name data must be defined separately before the actual A-number screening result is defined.
New Call Processing Behavior

This new longest match feature results in a new call processing behavior which enhances the basic analysis capability the following five situations:

- Longest Match in A-Number Analysis
- Longest Match in B-Number Analysis
- Dial Plan Changing
- Overlap Dial Plan Changing
- Ported Number Handling

**Longest Match in A-Number Analysis**

With analysis set to the new call processing capability, the following A-number analysis will be subject to longest matching where the new call processing result replaces the old one:

- ANNOUNCEMENT
- BLACKLIST
- CAUSE
- NEW_DIALPLAN
- A_NUM_DP_TABLE

**Longest Match in B-Number Analysis**

With analysis set to the new call processing capability, the following B-number analysis will be subject to longest matching where the new call processing result replaces the old one:

- ANNOUNCEMENT
- BLACKLIST
- CAUSE
- TERM_INFO
- NEW_DIALPLAN
- A_NUM_DP_TABLE
- ROUTE
- COND_ROUTE
- PERC_ROUTE
- MGCPDIALPKG
- E_PORTED_NUM
- E_ROUTE_NUM

**Dial Plan Changing**

With analysis set to the new call processing capability, dial plan changeover is not a forced action. Previously, a changing result with a ROUTE or ANNOUNCEMENT result would always force a dial plan change. Now change is optional and is carried out only if it is the longest match among the other results.
For example, in the original capability, if there is a number 1234, and the results at digits 12 was NEW_DIALPLAN, and at 123, the result was ROUTE, the dial plan was changed over. With the new capability, the status of the NEW_DIALPLAN and A_NUM_DP_TABLE results are “reduced” so that these can be longest matched against the other results. In this example, the call would be completed with the ROUTE result at digits 123, and there would be no dial plan changeover.

The new feature applies to all of the results listed below:

- For A-number analysis:
  - CAUSE
  - BLACKLIST
  - ANNOUNCEMENT
  - NEW_DIALPLAN
  - A_NUM_DP_TABLE

- For B-number analysis:
  - CAUSE
  - BLACKLIST
  - ANNOUNCEMENT
  - TERM_INFO
  - ROUTE
  - COND_ROUTE
  - PERC_ROUTE
  - MGCPDIALPKG
  - E_PORTED_NUM
  - E_ROUTE_NUM
  - A_NUM_DP_TABLE
  - NEW_DIALPLAN

**Overlap Dial Plan Changing**

When you are working with the analysis set to the new call processing capability, before processing a dial plan changeover, overlap calls are checked to see if analysis is complete. If it is not, then instead of forcing a dial plan changeover at this time, the system waits for digits. This allows for further digits to be analyzed in the search for a longer match. These extra digits might produce a different result, for example, ROUTE or ANNOUNCEMENT, which would then be executed instead of the change. This prevents the call from moving into the wrong dial plan and risking a failed call.

Following a valid change, an overlap call might still run out of digits and need more digits for the analysis to be complete. In that case, the analysis will return an appropriate indication to call control, forcing the call to wait for further digits. In overlap working, an initial address message (IAM) is delivered, and then further digits are delivered in subsequent address messages (SAM), which are received from the previous switch or line.

In addition, when the analysis capability is set to the new call processing capability, it changes back to the first dial plan rather than waiting for further digits in the current one. This allows the new analysis request to be processed as a completely new procedure and supports longest matching.
Ported Number Handling

When you are processing ported numbers, if the Cisco PGW 2200 Softswitch is a donor switch, the B-number analysis result E_PORTED_NUM is used. When detecting this result, the PGW does a times-10 database lookup with the called party number, and if it finds a match, a routing number is returned and this is added as a prefix to the called number. The number is then reanalyzed with the intention of finding a routing to the recipient switch.

With basic analysis capability, it was possible to provision a ROUTE result that could be used to route the call if the number was not matched in the same result set as the E_PORTED_NUM. In such cases, a ROUTE result either at a prior or later point in the digit tree will be used to complete the call.

With the new call processing capability, the E_PORTED_NUM and E_ROUTE_NUM results are now also subject to longest matching, along with the B-number analysis results CAUSE, BLACKLIST, NEW_DIALPLAN, ANNOUNCEMENT, TERM_INFO, ROUTE, COND_ROUTE, PERC_ROUTE, MGCPDIALPKG, and A_NUM_DP_TABLE. Consequently, a ported result displaces and removes any previous ROUTE result. Also if a ported result was configured with a default ROUTE result in the same result set, this latter ROUTE result would remove the E_PORTED_NUM and invalidate the porting.

To avoid this situation, routing data is preserved, provided that the ROUTE result is either before the E_PORTED_NUM result in the or is colocated with it in the same result set. Any route result at a later point in the digit tree overwrites and removes the ported result, as required with longest matching.

Reverting to First Dial Plan When There Are Insufficient Digits in Overlap

The Dial Plan Longest Match feature enables you to revert to the original dial plan when there are insufficient digits, and the existing dial plan changeover handling does not provide the flexibility you need throughout your dial plan structure.

The following examples show how the feature works.

Main dial plan
49 – Move to new dial plan 0001.
49123 – Move to new dial plan 0002.

0001 dial plan
491 – Route1

0002 dial plan
49123 – Route2

Example 1 – In the case of B-number 4912345, given the way the dial plans are provisioned, it is expected that the analysis in dial plan "Main" will result in a changeover to new dial plan 0002 from where the call will be routed. If the signaling mode is "Enbloc," this obviously works without any problem; however, in "Overlap" mode with certain call scenarios there can be a problem.

Example 2 – If the IAM delivers digits 49 and then the SAM delivers 12345, with the old functionality, 49 will result in a changeover to dial plan 0001 where the analysis would run out of digits. This would result in a wait for more digits within dial plan 0001. When digits 12345 are received in a SAM message, a new analysis attempt is made, and analysis continues from dial plan 0001, where the call is finally routed after matching 491 using Route1. The problem is that the call was routed via dial plan 0001, but the customer expected this to route via 0002 using the longest match.

To address this problem, the new overlap multiple dial plan functionality is altered so that if the analysis runs out of digits and waits for new digits, it changes back to the first dial plan. When a new analysis request is made (with further digits), it is treated as a new request and not as a continuation of the previous analysis.
With such functionality in place, in Example 2 after the IAM delivers digits 49, the dial plan is changed over to 0001 and runs out of digits. A wait for further digits is started, but this time the analysis changes back to dial plan "Main" before waiting. When the SAM message delivers digits 12345, the complete number 4912345 is sent to analysis, where it is treated as a new request. Starting in dial plan "Main," the longest match would be found against 49123, and it will change over to dial plan 0002 where the call would finally be routed.

**Result Set**

A result set is a grouping of result types that can be associated with an A-number analysis, B-number analysis, Pre-analysis, or Cause analysis. You can have only one result set for each digit string; however, you can have one or more result types in a result set. Each result set requires a unique name, and each result type within a result set also requires a unique result name. However, the result names do not need to be unique across result sets—it is the combination of result set name and result name that must be unique. The result set name and the result name can each be as many as 20 alphanumeric characters in length. Table C-3 in Appendix C, “Dial Planning Worksheets,” can be used to plan your result set.

When determining the result types for a result set, intermediate results have to be created before end-point results. For example, the intermediate result type SCREENING must be added before the end point result type MGCPDIALPKG. You can have as many intermediate result types in a result set as you want. However, once a result set has an endpoint analysis result type, that is the end of the result set.

Each result set supports only one occurrence of any of the result types. For example, the user cannot configure the result type ROUTE followed by another ROUTE in the same result set.

See [Provisioning the Result Set, page 4-58](#) for an example of MML commands used for provisioning the result set.

**Default Result Set**

The default result set allows you to configure an action to occur if no result sets have been associated with the call.

Only one default result set is allowed for each customer group ID. Creating a new default result type overwrites the previous default result type. Only one of the following result types is allowed for the default result set at any time:

- **BLACKLIST**—Analysis of the B-number reveals that it is on the black list and the call is released.
- **ROUTE**—Analysis of the B-number reveals that the call is to be routed elsewhere.
- **CAUSE**—Analysis of the B-number reveals that the call is to be released with a specified cause.

**Pre-analysis**

In Pre-analysis there are several serial stages (described in the following sections). After data in each stage is read, any accumulated results are put in a results “collection bin”. If duplicate results occur, the following result simply overwrites the previous result; at all times there is only one version of a particular result in the bin. The only exception to this is number modification in result types AMODDIG, BMODDIG, and RMODDIG, which are cumulative from stage to stage in Pre-analysis. Thus each digit modification string changes the number string and becomes the input to the next stage. At the end of all Pre-analysis stages, the accumulated results collected are saved and processed.
If a dial plan change is returned from Pre-analysis because of the NEW_DIALPLAN result type, then you change the dial plan to revert to Pre-analysis again using the new dial plan, starting analysis over again. For the NEW_DIALPLAN result type, the only analysis option is to return to Pre-analysis.

From Pre-analysis, A-number analysis is entered, and then B-number analysis is entered. In the last two stages, we traverse the digit analysis data and collect the results at various points, with the longest match applied to a selection of results. From the A-number analysis stage, a dialplan change can only provoke a return to Pre-analysis. However, from the B-number analysis stage a dialplan change can either revert to Pre-analysis or simply re-start B-number analysis (within the new dialplan). In these two stages, there is functionality, such as call screening or LNP, that requires early actions and sometimes early responses to call control.

### Calling Party Category Analysis

Calling party category (CPC) analysis is the first stage in Pre-analysis that enables analyzing the CPC value in the IAM or Setup message. For example, this would allow an emergency call arriving at the Cisco PGW 2200 Softswitch that has a well-known CPC value to be setup immediately.

In Pre-analysis, the CPC value from the IAM is analyzed from the CPC configured values, in which result sets are assigned to specific values. When a match occurs, a result set name is obtained. The result set name is used to read the result list to determine the action to be performed.

The CPC value, shown in Example 1-3, contains two fields: the CPC value and the result set name. The CPC value is matched by the CPC value received in the IAM or Setup message on the originating side.

#### Example 1-3 Calling Party Category Example

<table>
<thead>
<tr>
<th>CPC Value</th>
<th>Result Set Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>set1</td>
</tr>
<tr>
<td></td>
<td>...</td>
</tr>
</tbody>
</table>

**Note**
The CPC value is the MGC internal value and not the protocol-specific value. See the “CPC Values” section on page A-9 for a list of CPC values.

### Transmission Medium Requirement Analysis

Transmission medium requirement (TMR) analysis is the second stage in Pre-analysis that enables analyzing the TMR value in the IAM or Setup message. For example, this would allow the Cisco PGW 2200 Softswitch to set different media gateway bearer capabilities within the network.

In this Pre-analysis stage, the internal TMR value is matched against the provisioned TMR value. Match results are assigned to specific values. The TMR list, shown in Example 1-4, contains two fields: the TMR value and the result set name. The TMR value is matched by the TMR value received in the IAM or Setup message on the originating side with the TMR value. The match produced is used to read the results.

#### Example 1-4 Transmission Medium Requirement Example

<table>
<thead>
<tr>
<th>TMR Value</th>
<th>Result Set Name</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>...</td>
</tr>
</tbody>
</table>
Example 1-4  Transmission Medium Requirement Example

<table>
<thead>
<tr>
<th>TMR Value</th>
<th>Result Set Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>set1</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Note: The TMR value is the MGC internal value and not the protocol-specific value. See TMR Values, page A-17 for a list of TMR values.

A/B-number NOA and NPI Analysis

In software Release 9.4(1) nature of address (NOA) and numbering plan indicator (NPI) analysis of the A-number (CgPn) was added. As a result, there are two NOA tables, one for A-numbers and one for B-numbers. Similarly, there are two NPI tables, one for A-numbers and one for B-numbers.

NOA and NPI analysis is the third stage of Pre-analysis. NOA/NPI analysis is performed based on their respective provisioned values, as well as the NOA and NPI values contained in the incoming setup messages. The incoming NOA and NPI values are protocol dependent.

For the protocol-specific NOA and NPI values, and the unique mappings from the numerical values supported by each protocol to the internal call context values, see Appendix A, “NOA and NPI Codes, CPC and TMR Values.”

A/B-number Nature of Address

The NOA is used to define the actions to be taken based on the NOA value in the incoming call. There are three entries for the NOA MML command: the NOA value, NPI block value, and the result set name.

- **noavalue** — is the value specified in the NOA value column of the NOA.
- **npiblock** — is the value used to identify a unique NPI block in the NPI.
  - If the NPI block value is set to 0 by the user or not configured against an NOA value, no analysis is performed on the NPI. Pre-analysis is based only on the incoming NOA value.
  - If the NPI block value is set to any value other than 0, analysis is performed in the NPI block indicated by the NPI block value. Pre-analysis is based on both the NOA and NPI values.
- **setname** — is the result set name (setname=“set1”) used to associate a result set with the incoming NOA value.
  - If the result set name is not configured, then no action is taken.
  - If the result set name is configured, the action taken is based on the result types included in the specified result set.
Example 1-5  Nature of Address Example

<table>
<thead>
<tr>
<th>NOA Value</th>
<th>NPI Block Value</th>
<th>Result Set Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>set1</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>

For any NOA value that is configured, either an NPI block or a result set must be specified. Table C-6 in Appendix C, “Dial Planning Worksheets,” can be used for provisioning your NOA.

Note: The NOA value needs to be the MGC internal value and not the protocol-specific value. See Appendix A, “NOA and NPI Codes, CPC and TMR Values” for a list of NOA values.

A/B-number Numbering Plan Indicator

A separate NPI block is required for every non-zero entry in the NPI Block column of the NOA value that you want to associate with a result set.

- `npiblock` — is the value specified in the NPI Block column of the NOA.
- `blockvalue` — is the incoming NPI value as described in Appendix A, “NOA and NPI Codes, CPC and TMR Values.”

If a block value is not specified, all 16 entries (0 through 15) in the specified NPI block default to an empty result set name, so no action is performed.

- `setname` — is the result set name (setname="set1") associated with the incoming NPI value.

The result types included in the specified result set determine the call processing actions to be performed based on the incoming NPI value, as described in the “Result Set” section on page 1-70.

Example 1-6  Numbering Plan Indicator Example

<table>
<thead>
<tr>
<th>NPI Block</th>
<th>NPI Block Value</th>
<th>Result Set Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>set1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>set1</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>set1</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>set2</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>set3</td>
</tr>
<tr>
<td>1</td>
<td>5</td>
<td>set4</td>
</tr>
<tr>
<td>1</td>
<td>6</td>
<td>set1</td>
</tr>
<tr>
<td>1</td>
<td>7</td>
<td>set1</td>
</tr>
<tr>
<td>1</td>
<td>8</td>
<td>set1</td>
</tr>
</tbody>
</table>
Pre-analysis

Table C-7 in Appendix C, “Dial Planning Worksheets,” can be used for provisioning your NPI.

<table>
<thead>
<tr>
<th>NPI Block</th>
<th>NPI Block Value</th>
<th>Result Set Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>9</td>
<td>set2</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>set3</td>
</tr>
<tr>
<td>1</td>
<td>11</td>
<td>set4</td>
</tr>
<tr>
<td>1</td>
<td>12</td>
<td>set1</td>
</tr>
<tr>
<td>1</td>
<td>13</td>
<td>set1</td>
</tr>
<tr>
<td>1</td>
<td>14</td>
<td>set1</td>
</tr>
<tr>
<td>1</td>
<td>15</td>
<td>set5</td>
</tr>
</tbody>
</table>

Note

The NPI value needs to be the MGC internal value and not the protocol-specific value. See Appendix A, “NOA and NPI Codes, CPC and TMR Values” for a list of NPI values.

Transit Network Selection Analysis

Transit network selection (TNS) analysis is the fourth stage in Pre-analysis that enables analyzing the TNS values. For example, this would allow the Cisco PGW 2200 Softswitch to set different media gateway bearer capabilities within the network.

In this Pre-analysis stage, the internal TNS value is matched against the provisioned TNS value. The TNS value contains a digit string representing a carrierId. If the string is a match, then the associated result set is processed.

Example 1-7 Transit Network Selection Example

<table>
<thead>
<tr>
<th>TNS Value</th>
<th>Result Set Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>123</td>
<td>set1</td>
</tr>
<tr>
<td>223</td>
<td>set2</td>
</tr>
<tr>
<td>334</td>
<td>set3</td>
</tr>
</tbody>
</table>

NANP B-Number Normalization

The final stage in Pre-analysis is North American Numbering Plan (NANP) number normalization. NANP applies B-number normalization to intraLATA calls only for North American networks.

B-number normalization is required only if the number plan analysis (NPA) property contains the 3-digit string providing the NPA prefix for the associated trunk group. If the NPA property is empty, then B-number normalization is not required.

If B-number normalization is required, the NPA property value for the trunk group is prepended as a 3-digit number to the 7-digit B-number (NXX-XXXX). This creates a 10-digit B-number in the format NPA-NXX-XXXX.
Added Gateway Announcement Capability

The ANNOUNCEMENT result enables playing an early announcement to the MGC originating side before completing call routing. An application example might involve providing a “Welcome to the Network” message before setting up the call. This requires the associated gateway to have the capability to play announcements.

Note

Due to the nature of the connection, calls arriving by EISUP (H.323) or SIP protocols are not supported.

Applying an announcement is provisioned on the MGC at three levels: as an incoming trunk group property, as an A-number analysis result, or as a B-number analysis result. The trunk group property (PlayAnnouncement) provides the initial announcement identity and can be optionally over-written during A-number or B-number analysis.

Upon receiving a new call, the MGC analysis function carries out Pre-analysis, A-number analysis, B-number analysis, and routing. Early in the analysis function, the incoming trunk group property PlayAnnouncement is read and any provisioned announcement identity (integer value) is retrieved. If the PlayAnnouncement trunk group property is not configured, the property has a null or zero value indicating that no announcement action is required. If an announcement identity is retrieved, this indicates that an early or welcome announcement is to be played and that this data is stored locally before starting analysis, which is where the selection of announcement can be overridden.

If an ANNOUNCEMENT result is collected at Pre-analysis, A-number analysis, or B-number analysis, it is in addition to any announcement identity collected from the trunk group property. If the PlayAnnouncement trunk group property is present, it is applied. However, if one of the analysis areas also has provided an ANNOUNCEMENT result, then the result either overrides the trunk group property or negates the trunk group property by not applying the announcement for the incoming number. The basic rule is that the last analysis area determines the final result.

Note

If in overlap numbering mode, all digits must be received and analyzed before any announcement is played. This means that any overlap announcement call effectively goes from overlap to enbloc.

If the final ANNOUNCEMENT result has dataword4 set to indicate OFF, then no announcement is played. In this case, generic analysis completely removes the ANNOUNCEMENT result and the call status is determined by the other collected results.

Once generic analysis determines which announcement Id to use, this information is passed back to number analysis to perform the necessary action. The announcement data accompanies any other results being returned. The result from analysis varies according to the announcement type and according to both the final result and the delivered announcement data.

When the RSLT_ANNOUNCEMENT analysis result arrives, the accompanying data is examined. The first check is to determine if it is a remote or local (gateway) announcement. If the announcement is remote, then the previous functionality and handling are invoked. If the indicator is set to local, then a gateway announcement is required, the Times-Ten Announcement table is read, and the data is stored in readiness.

A check is made to determine the course of action. That is, the check determines if the announcement is the final action (play announcement and clear down) or an intermediate action (play announcement and continue processing), as indicated by dataword4.
With the course of action determined, the required call processing is initiated. The MGC sends a CRCX[M: recvonly] over MGCP to the originating side gateway to prepare the channel for announcement playing. Then the MGC sends an ACM signal back to the preceding switch. This is necessary to allow time for playing the announcement, without exceeding the ACM timers, and to prepare the speech path on the previous switches.

At this point, the announcement playing is started and a Notification Request (RQNT) is sent to the gateway. Upon receiving RQNT, the gateway plays the announcement and also positively responds to the command. The MGC times the announcement playing activity and awaits a Notify (NTFY) message from the gateway. Once announcement playing is complete, the gateway responds with the NTFY message, and the completion indication is passed back to the MGC.

The action the MGC takes at this point depends on the indication given by analysis in dataword4. If the indication is ANN_final, then the call is cleared down. However, if the indication is Ann_Interim, then processing continues with the remaining results and the ANNOUNCEMENT result is discarded. If the remaining results include a trunk group, then circuit selection takes place.

### Action If Announcement Is Disabled

You can allow an analysis ANNOUNCEMENT result to disable (switch off) an Announcement according to the A-number or the B-number. For example, the trunk group property PlayAnnouncement can be provisioned with an Announcement identity to play on all calls delivered to a specific trunk group. Then, later in A-number analysis, an ANNOUNCEMENT result is collected that switches this requirement off (dataword4 = 0). Effectively now there is no Announcement requirement, so Generic Analysis must discard the ANNOUNCEMENT result. When this occurs, there is no ANNOUNCEMENT result or Announcement data returned for further processing.

### Action When Announcement Is Enabled by Trunk Group and/or Analysis Result

If an ANNOUNCEMENT result type is received with dataword2 set to 1 (indicating a remote announcement), a Route is forwarded to the remote announcement server. Announcement information is also returned containing all the dataword information that has been collected.

If an ANNOUNCEMENT result type is received with dataword2 set to 0 (indicating a local announcement), then local gateway announcement handling is required. If in the accompanying Announcement data, dataword4 (AnnData) is set to 2 (Final announcement on), this indicates that the Announcement is required and the announcement is treated as a final action (that is, play the announcement and then clear down the call).

If an ANNOUNCEMENT result type is received with dataword2 set to 0 (indicating a local announcement), then gateway announcement handling is required. If in the accompanying announcement data, dataword4 (AnnData) is set to 1 (Interim announcement on), the announcement is required and is treated as an intermediate action (meaning that other results determine the final call processing action). For example, an early announcement case could occur if the MGC initiates the playing of an announcement and then routes the call forward. In this scenario, the announcement data is returned as optional data. However, the main result is set to reflect the end of analysis result retrieved (for example, TRUNK_GROUP, ANALYSIS_PERFORMED (nailed call), or ROUTE (Cause analysis result)).

**Note**

When dealing with these results, first determine if there is announcement data and take the appropriate action.
Times-Ten Database Announcement Table

Once it has been established that Tone or Announcement playing is required, the data must be retrieved to support the MGCP message request to the gateway. Within the MGC, the Times-Ten database holds this data in the Announcement table, which is represented in Table 1-11.

Table 1-11 Announcement Table Representation

<table>
<thead>
<tr>
<th>AnnId</th>
<th>gwType</th>
<th>playDuration</th>
<th>Repeat</th>
<th>Interval</th>
<th>locationString</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The Announcement table contains the following fields of data relevant to the Announcement package:

- **AnnId**—Indicates the announcement identity (or tone identity), which matches the announcement identity defined by the ANNOUNCEMENT result type. This is one of the two access keys for which the Announcement table is searched for a match. (4-digit integer).

- **gwType**—A string containing a value that is part of an enumerated set identifying the gateway type for this side of the call. This is the second access key used to read the Announcement table. (10 characters maximum).

- **playDuration**—Indicates the intended duration, in seconds, for which the announcement or tone is played. The default value is 60. Value range: 0 through 120.

- **repeat**—Indicates the number of times the announcement or tone is repeated; or indicates if it must be played continuously for the specified duration. A value of 0 indicates continuous playing. The default value is 1. Value range: 0 through 5 (4-digit integer).

- **interval**—Indicates the silence interval duration, in milliseconds, between replaying an announcement or tone. Default value: 3000. Value range: 0 through 5000 (4-digit integer).

- **locationString**—A string indicating to the gateway the audio file to load to enable announcement or tone playing. The string format varies according to the gateway type and its configuration. The string information is part of a URL string that the MGC sends by MGCP to the gateway. (Maximum length of string: 128-characters).

**Note**
The gateway is expected to support standard URL schemes for this notation (that is, file, http, or ftp), as described in RFC 1738.

For file or ftp versions, the data provisioned in this field is the required filename, because the gateway is expected to already know the directory location or ftp address.

For example:

- **Audiofile1.txt** (file or ftp version, gateway knows where to find the file)
- **MyName@host.dom/etc/Audiofile1.txt** (http version, gateway will retrieve the file from this location).

**Note**
Gateways do not support playDuration, repeat, and interval parameters at this time.
Number Analysis

A-Number Analysis

A-number analysis (see Figure 1-8) provides digit-by-digit analysis, and call screening that supports both blacklist and whitelist screening capability.

From the point of view of the Cisco PGW 2200 Softswitch, each digit received is processed separately. Each digit is processed through a tree-structured representation that is stored in the digit tree. Each digit tree allows analysis of the decadic digits 0 through 9 and the over-decadic (hexadecimal) 0 through 9 and digits A through F to be configured.

Figure 1-8  A-number Analysis Overview

Cause Analysis

Cause

Cause lists the cause codes generated when a call is rejected or cleared by the system (see Figure 1-9). The cause for release can be a result type (from either B-number analysis or Cause analysis) or a failure (generated during call processing). The cause codes are used as the release message for internal causes.

The two cause fields are the Location Block and Result Set Name, as shown in Example 1-8.

- The location block identifies a block of data specific to the network that the call is originated.
  If location block value is set to 0, no further analysis is performed based on the location.
- The result set name is used to associate a result set with a cause value.
  If a result set name is not configured, then no action is taken.
The Location Block and the Result Set Name cannot be provisioned the same time. For more information on provisioning causes, see Chapter 5, “NUMAN: Commands for Provisioning Dial Plan Components” of the Cisco PGW 2200 Softswitch Release 9 MML Command Reference.

Each location block can hold up to 16 entries. Each entry identifies a result set. A location block entry must be configured with a result set name other than null (0). For information on location blocks and location block entries, see the “Location” section on page 1-79.

If both the location block and the result set name are set to null, no analysis is performed.

Note

The RETRY_ACTION result type allows you to select the default cause or re-edit the cause during dial plan creation. The default cause is used as before. However, if you desire to specify a retry action, you must enter a retry value for dataword1.

**Figure 1-9  Cause Analysis Overview**

![Cause Analysis Overview Diagram]

**Example 1-8  Cause Example**

<table>
<thead>
<tr>
<th>Cause Value</th>
<th>Location Block</th>
<th>Result Set Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>set1</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

See Appendix B, “Cause and Location Codes,” for a list of the cause codes for the protocol variants. Table C-8 in Appendix C, “Dial Planning Worksheets,” can be used to plan the Cause values.

Note

The cause and location values used here are the internal values, not those seen in a REL message. See Appendix B, “Cause and Location Codes.”

**Location**

Location is used to identify an associated result set, as shown in Example 1-9. Location is accessed from the cause value through the locationblock value. The locationblock value refers to a block of up to 16 entries (0 through 15).
There can be multiple location block entries in one location block. The *blockvalue* specifies an offset into the specified location block. You can associate an action with the specific *blockvalue* by setting the result set name (*setname*) at the specified offset in the location block.

On the Cisco PGW 2200 Softswitch, you always provision internal values for cause and location codes. During cause analysis, for different protocols, the Cisco PGW 2200 Softswitch do two mappings, mapping received cause and location codes to internal cause and location codes, and mapping internal cause and location codes to protocol-specific cause and location codes. Then, the Cisco PGW 2200 Softswitch uses internal cause and location codes to determine further actions.

**Note**

For details on cause and location code mappings for different protocols, see Appendix B, “Cause and Location Codes”.

If the cause code in the Release message has a location significance, the Release message has a value in the location indicator. The Cisco PGW 2200 Softswitch maps the received cause code to an internal cause code value for this specific protocol. The internal cause code value identifies the location block. The Cisco PGW 2200 Softswitch maps the location indicator to an internal location value. Then it uses the internal location value as an index into the location block to identify the location block entry. If that entry exists, the Cisco PGW 2200 Softswitch uses the result set provisioned in that entry as result actions.

**Figure 1-10** gives an example for cause analysis on cause codes and locations. It assumes that the call is using ANSI SS7 protocol.

In this example, a user associates the result set, set2, with the internal cause code 12. The location block value for the cause code 12 entry is set to 0. The user sets the location block to 2 for the internal cause code 34. In Location Block 2, the user associates the result set, set3, with the location block value 6.

**Note**

The *blockvalue* in numan-add:location should be one less than the intended internal value. For detailed provisioning procedures, see the “Location Mapping” section, in Chapter 5, Adding System Components with MML, of *Cisco PGW 2200 Softswitch Release 9 Provisioning Guide (through Release 9.7)*.

If the received cause code in the release message was mapped to the internal cause code 12, the Cisco PGW 2200 Softswitch uses set2 as result actions. If the mapped-to internal cause code is 34, the Cisco PGW 2200 Softswitch identifies the location block 2. Then the Cisco PGW 2200 Softswitch maps the location indicator 7 to the internal location value LOCATION_INTERNALATIONAL according to Table B-33, Protocol-specific Release Cause Location Values. Because the internal location value minus 1 equals 6, the Cisco PGW 2200 Softswitch identifies set3 as result actions.
See Appendix B, “Cause and Location Codes,” for a list of the location codes for the protocol variants. Table C-9 in Appendix C, “Dial Planning Worksheets,” can be used to plan the Location values.

Example 1-9  Location Example

<table>
<thead>
<tr>
<th>Location Block</th>
<th>Location Block Value</th>
<th>Result Set Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>set1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>set2</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>set1</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>set2</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>set3</td>
</tr>
<tr>
<td>1</td>
<td>5</td>
<td>set4</td>
</tr>
<tr>
<td>1</td>
<td>6</td>
<td>set3</td>
</tr>
<tr>
<td>1</td>
<td>7</td>
<td>set1</td>
</tr>
<tr>
<td>1</td>
<td>8</td>
<td>set1</td>
</tr>
<tr>
<td>1</td>
<td>9</td>
<td>set2</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>set3</td>
</tr>
<tr>
<td>1</td>
<td>11</td>
<td>set4</td>
</tr>
<tr>
<td>1</td>
<td>12</td>
<td>set1</td>
</tr>
<tr>
<td>1</td>
<td>13</td>
<td>set1</td>
</tr>
<tr>
<td>1</td>
<td>14</td>
<td>set1</td>
</tr>
<tr>
<td>1</td>
<td>15</td>
<td>set5</td>
</tr>
</tbody>
</table>
Dial Plan Selection

To meet the requirements of multiple dial plans, it is necessary to identify dial plan identity strings referenced by an integer value, as shown in Example 1-10. This is called a dial plan selection table and is located within the dial plan.

During call processing, the NEW_DIALPLAN result provides the CustGrpId value in dataword1 that is used to read the value and retrieve a new dial plan identity for continuing analysis.

Example 1-10 Dial Plan Selection

<table>
<thead>
<tr>
<th>CustGrpId</th>
<th>Dial Plan Id</th>
</tr>
</thead>
<tbody>
<tr>
<td>t001</td>
<td>N001</td>
</tr>
<tr>
<td>t002</td>
<td>N002</td>
</tr>
<tr>
<td>t003</td>
<td>N003</td>
</tr>
</tbody>
</table>

A-Number Dial Plan Selection

The dial plan selection table lets you select a new dial plan based on the incoming CustGrpID and the full A-number. As shown in Example 1-11, this requires three fields, a string CustGrpId, an A-number string, and a dial plan identity string.

Example 1-11 A-Number Dial Plan Selection

<table>
<thead>
<tr>
<th>CustGrpId</th>
<th>A-number</th>
<th>New DialPlanId</th>
</tr>
</thead>
<tbody>
<tr>
<td>t001</td>
<td>02087568111</td>
<td>dp07</td>
</tr>
<tr>
<td>t002</td>
<td>01444234567</td>
<td>dp07</td>
</tr>
<tr>
<td>t003</td>
<td>01494333221</td>
<td>dp08</td>
</tr>
</tbody>
</table>

With the multiple dial plan capability, it is quite possible that some dial plans may be accessed only as the result of call processing using a previous dial plan. This means that some dial plans might not be associated with a trunk group or sigpath that requires another list, which provides a complete list of all valid dial plans that are loaded at startup.

Multiple Dial Plan Result Types

This section provides information on the multiple dial plan functionality.

During A-number Analysis

- Identify from A-number analysis that a new dial plan should be selected.
- Identify from A-number analysis that a new dial plan could potentially be selected, basing the decision on the full analysis of the A-number.
- Support optional A-number modification, B-number modification and A-number and B-number NOA modification. They are processed before any dial plan change.
- Before changing dial plans, any other results obtained must be processed.
• Following a dial plan change, analysis resumes within the new dial plan at the Pre-analysis stage.

**During B-number Analysis:**
• Identify from B-number analysis that a new dial plan should be selected.
• Support optional A-number modification, B-number modification and A-number and B-number NOA modification. They are processed before any dial plan change.
• Before changing dial plans, any other results obtained must be processed.
• When invoking a new dial plan from B-number analysis, support the capability to restart B-number analysis within the new dial plan.
• When invoking a new dial plan from B-number analysis, support the capability to restart analysis from the Pre-analysis stage within the new dial plan.

**General Objectives:**
• Maximum of ten dial plan changes per call. The number of dial plan changes is not provisionable.
  If this limit is reached, call processing will complete within the current dial plan.
• A dial plan change result signals the end of analysis within the current dial plan.

### Dial Plan Features

#### Call Screening

Call screening is one type of analysis performed on the calling number (A-number) and the called number (B-number) to determine if a call is to be accepted or rejected. The Cisco PGW 2200 Softswitch supports whitelist call screening that allows listed numbers and blocks all others, and blacklist call screening that blocks listed numbers and allows all others.

<table>
<thead>
<tr>
<th>A-Number Status</th>
<th>Whitelist Action</th>
<th>Blacklist Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>A-number listed</td>
<td>Call completed</td>
<td>Call terminated</td>
</tr>
<tr>
<td>A-number not listed</td>
<td>Call terminated</td>
<td>Call completed</td>
</tr>
</tbody>
</table>

**Note**
Screening is limited to 20 digits.

Either whitelist or blacklist call screening can be triggered from either A-number analysis or B-number analysis; however, only the calling number is screened. If screening is triggered from A-number analysis, the calling number is screened regardless of the number dialed. If screening is triggered from B-number analysis, the calling numbers allowed or blocked are limited to those associated with a specific service name. The screening is always performed on the calling number (A-number), regardless of which type of number analysis triggers the SCREENING result type.

Call screening verifies that a call can be completed. You can provision whitelists that specify allowed numbers and blacklists that specify blocked numbers, as shown in Table 1-12.
Whitelist or blacklist screening triggered by A-number analysis or B-number analysis results in four different ways to trigger call screening:

- **Whitelist Screening Triggered by A-Number Analysis**
- **Blacklist Screening Triggered by A-Number Analysis**
- **Whitelist Screening Triggered by B-Number Analysis**
- **Blacklist Screening Triggered by B-Number Analysis**

### Whitelist Screening Triggered by A-Number Analysis

In whitelist screening triggered by A-number analysis, the call is completed if the A-number digit string in the dial plan is included in the AWhite (whitelist) screening file. The call is terminated if the A-number digit string is not listed in the whitelist screening file. Figure 1-11 is an example of whitelist screening triggered by A-number analysis.

![Figure 1-11 Whitelist Screening Triggered by A-Number Analysis](image)

In the dial plan, the calling number digit string “301” is linked to the SCREENING result type during A-number analysis. When a call is placed, any calling number beginning with the digit string “301” is screened to see if it is included in the AWhite (whitelist) screening file.

In this example, if the calling number is 3016484444, the call is not completed because that number is not included in the AWhite screening file. However, if the calling number is 3016485555, the call is completed because that number is included in the AWhite screening file.

For the detailed procedure for building or adding to the AWhite screening file, see the Adding Screening Lists (SCREENING), page 4-23.

---

**Note**

The called number (B-number) has no effect on the call. In fact, the B-number does not have to appear in the dial plan at all when doing whitelist screening triggered by A-number analysis.
Blacklist Screening Triggered by A-Number Analysis

In blacklist screening triggered by A-number analysis, the call is terminated if the A-number digit string in the dial plan is included in the ABlack (blacklist) screening file. The call is completed if the A-number digit string is not listed in the blacklist screening file. Figure 1-12 is an example of blacklist screening triggered by A-number analysis.

Figure 1-12 Blacklist Screening Triggered by A-Number Analysis

In the dial plan, the calling number digit string “301” is linked to the SCREENING result type during A-number analysis. When a call is placed, any calling number beginning with the digit string “301” is screened to see if it is included in the ABlack (blacklist) screening file.

In this example, if the calling number digit string is 3016484444, the call is terminated because that number is included in the ABlack screening file. However, if the calling number digit string is 3016485555, the call is completed because that number is not included in the ABlack screening file.

For the detailed procedure for adding to the ABlack screening file, see the “Adding Screening Lists (SCREENING)” section on page 4-23.

Note

The called number (B-number) again has no effect on the call. The B-number does not have to appear in the dial plan at all when doing blacklist screening triggered by A-number analysis.

Whitelist Screening Triggered by B-Number Analysis

Screening triggered by B-number analysis is not as straightforward as screening triggered by A-number analysis. In screening triggered by A-number analysis, there can be only one A-number whitelist or blacklist per dial plan and a given A-number can appear only once in either screening file. In screening triggered by B-number analysis, also there can be only one B-number whitelist or blacklist per dial plan; however, these screening lists contain only A-number digit strings. Any given A-number digit string can appear multiple times in the BWhite screening file or the BBBlack screening file if the A-number is associated with a different service name at each appearance.

In whitelist screening triggered by B-number analysis, the called number triggers the SCREENING result type. Dataword1 contains a value of 1, indicating whitelist screening is requested, and dataword2 contains a recognized service name that is associated with the A-number digit string.
Figure 1-13 is an example of whitelist screening triggered by B-number analysis.

Note
Despite the fact that B-number analysis triggers the SCREENING result type in the following examples, remember that screening is always performed on the calling number (A-number) digit string.

**Figure 1-13  Whitelist Screening Triggered by B-Number Analysis**

In the dial plan, the called digit string “7034” is linked during B-number analysis to a result set that includes the SCREENING result type, which is also associated with the Washington service. When a call is placed to a number that begins with the digit string “7034,” the calling number is screened. If the calling number is included in the BWhite (whitelist) screening file and it is associated with the Washington service, the call is completed; otherwise, the call is terminated.

In this example, when a customer dials a number that begins with the digits “7034,” the calling number (3016484444) is screened and the call is terminated because this calling number is either not included in the BWhite screening file or it is included, but it is not associated with the Washington service. However, if the calling number were 3016485555, the call would be connected because that number is included in the BWhite screening file and it is associated with the Washington service.

To add numbers to a BWhite screening file, see the “Adding Screening Lists (SCREENING)” section on page 4-23.

**Blacklist Screening Triggered by B-Number Analysis**

In the case of blacklist screening triggered by B-number analysis, if the called number (B-number) digit string is associated with a service name and the calling number (A-number) is included in the BBlack (blacklist) screening file and it is associated with the same service name, the call is terminated; otherwise, the call is connected. Figure 1-14 is an example of blacklist screening triggered by B-number analysis.
In the dial plan, the called digit string “7034” was linked to the SCREENING result type during B-number analysis. When the call is placed, the calling number 3016485555 is either not included in the BBlack screening file or it is included, but not associated with the Washington service, so the call is connected. However, if the calling number is 3016484444, the call is terminated because that number is included in the BBlack (blacklist) screening file and is associated with the Washington service.

To add numbers to a BBlack screening file, see the “Adding Screening Lists (SCREENING)” section on page 4-23.

Redirecting Number Screening

Caution

The redirecting number screening capability has no effect on the provisioning of the A-number screening and analysis described previously; however, this screening capability is not backward compatible with Cisco PGW 2200 Softswitch releases earlier than Release 9.2(2).

Redirecting number screening is designed to augment, not replace, screening of the original A-number by introducing screening of the redirecting number. Redirecting number screening allows you to specify whether redirected calls are screened by using the original A-number or the redirecting number, which was the original B-number when the call was initiated, as shown in Figure 1-15.

For redirected calls, the calling number parameter contains the A-number of the station that originated the call, the redirecting number parameter contains the number of the station that redirected the call, and the called number parameter contains the number of the station to which the call is redirected.

If a succeeding switch should determine that a redirected call is to be subjected to A-number screening, it uses the number contained in the redirecting number parameter in the A-number screening process.
A-number screening for redirected calls can vary from customer to customer, so an office-based or switch-based parameter is required to specify the numbers that are used for A-number screening. The XEConfigParm.dat file contains office-wide or switch-wide parameters in the Cisco PGW 2200 Softswitch software, including the MDLANumberScreening and RedirectingATree parameters.

The default value of the MDLANumberScreening parameter (0) invokes the standard A-number screening on the number in the calling number parameter, regardless of whether the call is redirected or not. No screening is done on the number in the called party parameter or the redirecting number parameter.

Redirecting number screening is enabled by setting the MDLANumberScreening parameter to 1 in the XEConfigParm.dat file.

- If the original A-number screening was invoked by an A-number analysis SCREENING result, only the original A-number is screened.
- If the original A-number screening was invoked by a B-number analysis SCREENING result, either the original A-number or the redirecting number is screened, dependent on the presence of the redirecting number data in the received setup message.

When the redirecting number is screened, the setup message is returned to A-number screening where it is re-screened to determine whether the call can be completed based on the redirecting number instead of the original calling number. If the original B-number, now the redirecting number, can make calls to the new B-number, the call is completed.

Redirecting Number Screening in A-Number Analysis

This feature requires setting the RedirectingATree parameter in the XEConfigParm.dat file on initial configuration. When the RedirectingATree property is set, only BLACKLIST (screening criteria CLI) or SCREENING result types should be provisioned in the dial plan.

- If BLACKLIST is provisioned, the redirecting number is used for BLACKLIST functionality.
- If SCREENING is provisioned, the redirecting number is used for SCREENING functionality.

If result types other than BLACKLIST or SCREENING are encountered in the dial plan, an alarm (RedirectingNbrFail) is generated and processing continues with the normal Adigittree decode of the original A-number.

If the RedirectingATree property is not set, or there is no redirecting number present, then this stage is skipped and processing continues with the normal Adigittree decode of the original A-number.

If the result types digit modification (AMODDIG or BMODDIG) or number type (A_NUMBER_TYPE or B_NUMBER_TYPE) is encountered when a redirecting number is used during A-number analysis, no modification is performed on the redirecting number. In addition, the A-number analysis does not set the screening indicator field, because this field is not applicable for the redirecting number.

European Local Number Portability

European local number portability service (European LNP) is implemented in the Cisco PGW 2200 Softswitch for use only in the European-Middle East-Africa (EMEA) region. This service offers subscribers the ability to resign their subscription with their current service provider (donor network) and register with another service provider (recipient network) without changing their telephone number, location, or services.
There are three types of local number portability:

- **Service Portability**—Allows subscribers to retain their telephone number when changing from one type of service to another.
- **Service Provider Portability**—Allows subscribers to retain their telephone number when changing from one service provider to another.
- **Location Portability**—Allows subscribers to retain their telephone number when moving from one location to another. In this case the subscriber may or may not change service providers or services.

The subscriber’s telephone number, which identifies the subscriber, is either a service number or a logical number; however, it does not identify the subscriber’s service provider or provide any other information regarding the service provider.

To implement European LNP capability, an “onward routing” architecture is used. In this architecture type, only the donor network/switch has the number portability information and thus the complete address of the recipient network/exchange.

### European LNP Call Scenarios

Within the EMEA region, the Cisco PGW 2200 Softswitch is able to handle ported calls in the capacity of donor switch, transit switch, or recipient switch. The following paragraphs describe call scenarios from each of these network perspectives. SS7 signaling is required for number portability services.

**Donor Network**

If the Cisco PGW 2200 Softswitch is in a donor switch capacity when the called number is ported, it will have the result type E_PORTED_NUM provisioned in the dial plan which is retrieved during its digit analysis decode. This result type would only be provisioned when the Cisco PGW 2200 Softswitch is in a donor switch capacity.

Retrieval of the E_PORTED_NUM result type is an indication to read the ported number. This can be carried out only if all digits have been received. With en bloc numbering, the processing can continue directly; and with overlap numbering, the Cisco PGW 2200 Softswitch must await further digits until sending is complete.

**Note**

Screening is limited to 20 digits.

In this way the E_PORTED_NUM result type can be provisioned at the area code level, but the ported number cannot be interrogated until the complete number is received.

Calls to a ported number are routed to the recipient network using a unique routing number (called a Network Identification Code (NIC)), which is retrieved from the number portability (NP) database on the donor switch. A concatenated address is used, in which the routing number is prefixed to the B-number to transmit it to the next node. The length of the routing number is fixed within each country, but can vary from country to country.

The E_PORTED_NUM result type has one dataword that is used to enable the removal of any normalization prefix digits before prefixing with the routing number.

During B-number analysis the LNP call processing that takes place for a ported number in a donor network is as follows:

1. The donor switch receives an SS7 IAM containing a B-number (that is, called party number or CdPN).
2. After all the digits have been collected and the E_PORTED_NUM result type is encountered during B-number analysis, the Ported Number table (PORTTBL) is accessed to determine if the B-number has been ported to another network.
   a. If the B-number is not found in the ported number list, the call is routed according to the B-number.
   b. If the B-number is found in the ported number list, the donor switch performs an NP query to its local NP database to retrieve the recipient network routing number.
      If no routing information is defined for the routing number in the local NP database, the internal cause code is set to 51 (unallocated number) and the call is subjected to Cause analysis.
   c. If the ported number list could not be accessed due to a database error, the call is handled based on the database access error action set in the XEConfigParm.dat file:
      • If set to continue, the call is treated as a non-ported call and is routed according to the B-number.
      • If set to block, the internal cause code is set to 50 (temporary failure) and the call is subjected to Cause analysis.

3. The donor switch prefixes the routing number to the B-number and modifies the NOA parameter to Network Routing Number concatenated with the called party number (RN+CdPN).
4. B-number analysis is re-entered from the beginning using the concatenated address (RN+CdPN) to route the call onward towards the recipient network.

Transit Network
The LNP call processing that takes place for a ported number in a transit network is as follows:
1. The transit switch receives an SS7 IAM containing the new NOA with a B-number (B-number) and a routing number prefixed to it (RN+CdPN).
2. Early B-number analysis recognizes the new NOA. The Cisco PGW 2200 Softswitch performs standard B-number analysis.
3. The transit switch determines the route leading to the recipient network based on the RN.
   a. If a route is defined, Route Analysis is performed to route the call. The B-number with the prefixed routing number is transmitted transparently
   b. If a route is not defined for the Routing Number, the Cause code is set to 51 (unallocated number) and the call is subjected to Cause Analysis.
4. The transit network passes the B-number as it was received in the incoming IAM.

Recipient Network
The LNP call processing that takes place for a ported number in a recipient network is as follows:
1. The recipient switch receives an SS7 IAM containing the new NOA with a B-number and a routing number prefixed to it (RN+CdPN).
2. Early B-number analysis recognizes the new NOA and performs standard B-number analysis.
3. If the result type E_ROUTE_NUM is encountered, analysis waits until all the digits have been collected, then strips the leading digits (the routing number) from the B-number as defined in dataword1, modifies the NOA to national, and reads the Number Termination table (TERMTBL) using the B-number to retrieve the route(s) to the recipient switch.
   a. If no routing is defined in Number Termination, the internal cause code is set to 51 (unallocated number) and the call is subjected to Cause analysis.
   b. If Number Termination cannot be accessed due to a database error, the call is handled based on the database access error action set in the XEConfigParm.dat file:
      • If set to continue, the call is treated as a non-ported call and is routed according to the B-number.
      • If set to block, the internal cause code is set to 50 (temporary failure) and the call is sent to Cause analysis.

Advice of Charge

The Advice of Charge (AOC) feature (currently supported protocol variants are ISUPV2_GERMAN, Q761_INDIA, and ISUPV2_POLISH) is controlled by the ingress trunk group property AOC Enabled. The flat rate charging capability provides a simple fixed rate tariff feature that co-exists with the AOC variable rate charging capability.

Selecting the charging scheme used is determined by the content of the Charge Data Type field (dataword4) obtained from the CHARGE result type in the dial plan.

Access to the Charge list is made using the charge origin, charge destination, and day-of-week values. Charge origin and charge destination are obtained from A/B number analysis and passed to the CDR Manager by the call context. The date and day-of-week are read using an internal function. Upon retrieving the date and day-of-week values, the Holiday list is checked using another internal function to determine if the day of the week value is to be overwritten by a holiday value.

Three input parameters (charge origin, charge destination, and day-of-week) are passed into the function and a list of tariff IDs and change-over times is returned.

Access to the Charge list is made by retrieving the tariff descriptor, which is then used to access the Tariff list to obtain the tariff rate and scale factor. If a split-day tariff descriptor is obtained, it is separated into Tariff/Switchover-Time descriptor pairs before accessing the Tariff list using the current tariff descriptor to obtain the tariff rate and scale factor.

For split-day tariffs, the tariff rate and scale factor corresponding to the next time period and the switchover time to the next tariff rate are also retrieved from the Charge list and this information is also sent in the initial CRGT message. The data is made available to the ingress protocol by placing it in call context containers and this initial tariff data is also written to the CDR. The CDR Manager raises a minor alarm for any data access failure.

Two timers are used to prompt the CDR Manager that the call is about to change to the next tariff rate. Upon expiration of either timer, the charging lists are accessed again using the relevant tariff descriptor to retrieve subsequent tariff information, which is placed in the call context.

The relevant tariff descriptors are:

• The CRGT message indicating this changeover is sent 12 minutes before the switchover is to occur.
• The tariff duration specifies the duration until the next sub-tariff takes effect. Currently, the duration field in the CRGT message is hard coded to unlimited.
If the internal process receives subsequent charging information notifications from the CDR, the internal process outputs a signal to the ingress protocol indicating tariff switchover has occurred and subsequent tariff data is available to be sent.

**Provisioning AOC**

Three lists to be configured and loaded for AOC are:
- Charge
- Tariff
- chargeHoliday

Provisioning AOC is accomplished in the following stages:
1. Defining charge origins.
2. Defining charge destinations in the B-number lists.
3. Defining the customer specific holidays in the Holiday list.
4. Creation of the Charge list and population of the required tariff IDs for the identified charge origin/destination/day-of-week combinations.
5. Population of the tariff rates within the Tariff list.
6. Enabling AOC against ingress trunk groups or signal paths (sigpaths).

For detailed information on the AOC provisioning commands, see the “Provisioning Advice of Charge” section on page 4-31.

**Defining Charge Origins**

Charge origins are integer values from 1 through 9999. The charge origin can be assigned as a property against the trunk group or sigpath, a result type in the A-number analysis, or an entry in the CLI Charge Origin list. The user decides the charge origin value to be used. Typically, these numbers are incrementally assigned when planning the data build. However, the user can choose to use any valid value at any time.

- Trunk group or sigpath property
  - The property ChargeOrigin resides in the properties.dat file. It can be assigned to either the trunk group or the sigpath property, for example: TG-2.ChargeOrigin = 123.

- A-number result
  - In the result type CHARGEORIGIN, only the first dataword (dataword1) is significant and carries the charge origin value. This means that additional digit analysis continues past this result type. See the “Operation of Intermediate Result Types” section on page 1-16 for more information on intermediate result types and how they function in dial plans.

- CLI charge origin list

**Defining Charge Destinations**

The existing CHARGE result type includes the option of returning a charging destination. This is achieved by using the value in the ChargeDataDiscriminator field. The format of the accompanying data is described in the “CHARGE” section on page 1-25.
Creating the Holiday List

Each row in the chargeHoliday list is referenced by the DATE (a string value), which is composed of three integers representing the year, month, and day-of-week. The retrieved row entry is an integer in the range of 8-10, which displays as HOL1 (8), HOL2 (9), and HOL3 (10).

Creating the Charge List

You can now create Charge lists. In addition, you can add, edit, and delete rows within the Charge list. Each row in the list is referenced using three keys: charge origin, charge destination, and the day of the week. The charge origin range is from 1 through 9999, the charge destination range is from 1 through 9999, and the day of the week range is from Monday (1) through Sunday (7), HOL1 (8), HOL2 (9), and HOL3 (10).

Note

The only mandatory value is charge destination. Charge origin and day of the week, if absent, are set to 0 in the list row entry.

If the charge origin is absent, the entered row refers to all origins for that destination (unless explicitly entered in another row). Similarly, if the day of the week is absent, it refers to all days of the week not otherwise explicitly entered.

Creating the Tariff List

The Tariff list contains all required tariff rates and scale factors. Each row is referenced by a tariff ID, which call processing obtains by accessing the Charge list. The retrieved row entry contains the tariff rate and the scale factor.

Activating AOC

This capability is controlled by the property AOCEnabled (1 = enabled, 0 = disabled). To reduce alarms due to charging information pointing to unpopulated lists when provisioning AOC, it is advisable that AOC be disabled (AOCEnabled parameter set to 0) on the relevant trunk groups until the Charge list has been correctly updated.

PRI AOC supplementary services provisioning is accomplished in following stages:

Defining charge origins—Can be assigned to trunk groups or signaling paths, area codes (in the A-digit trees), or in a CLI Charge Origin table.
Defining charge destinations in B-number tables.
Defining customer-specific holidays using the Charge Holiday table.
Creating the PRI Charge table and populating the required tariff IDs for the identified charge origin, charge destination, and day of week combination.
Populating the tariff rates within the PRI Tariff table.
Configuring AOCInvokeType against the trunk groups.
Configuring AOCDefaultTariffId against the trunk groups.
Configuring AOCDMinPeriodicDuration against the signaling paths.
Enabling AOC against the ingress trunk groups or signaling paths.
AOC Generation for PRI

The Advice of Charge (AOC) Generation over PRI feature enables the Cisco PGW 2200 Softswitch to support the Advice of Charge Supplementary Service as a charge determination point for phones that are connected to Private Branch Exchange (PBX) switches. The Cisco PGW 2200 Softswitch determines the applicable tariff rates and sends AOC-S, AOC-D, and AOC-E messages over Primary Rate Interface (PRI) links, as defined in ETS 300 182.

AOC is a group of supplementary services that provides the served user with usage-based charging information. The MGC supports all three AOC supplementary services:

- Charging information at call setup time (AOC-S): The AOC-S supplementary service enables a user or subscriber to receive charging information about the charging rates at the call setup time; or at the latest, at call connection and also during the call, if charging rates change.

- Charging information during the call (AOC-D). The AOC-D supplementary service enables the user or subscriber to periodically receive the charging information on the recorded charges for a call during the active phase of the call. The MGC provides the charging information to the served subscriber or user in a facility message and also in a control message when clearing the call.

- Charging information at the end of the call (AOC-E). The AOC-E supplementary service enables the user or subscriber to receive charging information on the recorded charges for a call when the call is terminated. When clearing the call, the MGC provides the charging information to the served subscriber or user in a control message.

When the MGCP 1.0 on the PGW 2200 feature is enabled in the MGC, AOC is applicable only for the user or subscriber when that person is connected to the originating network. Also, if AOC is enabled and configured, the charging information for any of the three supplementary services can be provided for:

- All calls (AOCInvokeType is set to 2) received from the originating network on a configured trunk, which is referred to as “all calls”;

- A specified call (AOCInvokeType is set to 1) on the originating network, after the subscriber or user has requested the MGC to provide the charging information.

The Cisco PGW 2200 Softswitch activates an AOC supplementary service on a per call basis when the user has included in the SETUP message a Facility Information element containing a ChargingRequest invoke component. The ChargingRequest invoke component indicates the AOC supplementary service to activate. Each AOC supplementary service is activated independently. Thus, one, two, or three AOC activations can occur in the same SETUP message.

Upon receiving the ChargingRequest invoke component (depending on the parameters configured), the Cisco PGW 2200 Softswitch activates the requested AOC supplementary service and acknowledges the request by returning a ChargingRequest return result component within a Facility Information element in a subsequent call control message to the subscriber or user indicating that “chargingInfoFollows.”

If a proper per call configuration does not exist for the call requesting an activation, and the Cisco PGW 2200 Softswitch cannot activate the requested AOC supplementary service, the Cisco PGW 2200 Softswitch sends a Facility Information element. The element includes the requested type of AOC supplementary service (AOCSCurrency, AOCDCurrencyUnit, AOCEDChargingUnit, and so on), the invoke component indicating that “NoChargingInfoAvailable” or another error from the General Error list to the subscriber or user.

The Cisco PGW 2200 Softswitch continues processing the call normally, even if the AOC supplementary service requested was not activated. The subscriber or user must take the action that is suitable for the call.
Additionally, AOC over PRI can be configured to charge rates for a specified duration of the call, followed by a flat rate that is charged for the remaining call duration. This change to AOC allows for tariff changes during the call that are based on the duration of the call. For example, support for one tariff rate charge rate for the first 2 minutes of the call (that is, 40 units for the first 2 minutes or less), followed by a standard tariff rate of 10 units per minute for the remaining call duration. The remaining call tariff rate may or may not change based on the time of day or the day of the week.

Provisioning PRI AOC Supplementary Services

PRI AOC supplementary services provisioning is accomplished in following stages:

- Defining charge origins—Can be assigned to trunk groups or signaling paths, area codes (in the A-digit trees), or in a CLI Charge Origin table.
- Defining charge destinations in B-number tables.
- Defining customer-specific holidays using the Charge Holiday table.
- Creating the PRI Charge table and populating the required tariff IDs for the identified charge origin, charge destination, and day of week combination.
- Populating the tariff rates within the PRI Tariff table.
- Configuring AOCInvokeType against the trunk groups.
- Configuring AOCDefaultTariffId against the trunk groups.
- Configuring AOCDMinPeriodicTimerDuration against the signaling paths.
- Enabling AOC against the ingress trunk groups or signaling paths.

Charge Table

The Metering Pulse Messages (MPM) Support Feature is described in the following sections.

The Metering Pulse Feature enables the handling of meter pulse message pass through, modification, and generation. Billing information is derived from and provided to the billing mediator using Call Detail Records (CDRs).

This feature enhances the following two main functional areas of the MGC:

- Additional charging requirements—The MGC uses one or more of the following criteria to calculate charge tariff determination:
  - Incoming trunk group
  - Calling party number (also referred to as A-number)
  - Called party number (also referred to as B-number)
  - Calling Party Category (CPC)
  - Transmission Medium Requirement (TMR)

Charging information in the form of meter pulse messages (MPMs) is sent to the PSTN at call answer and/or periodically thereafter, depending on the tariff data provisioned in the Cisco PGW 2200 Softswitch. The sent MPMs are also recorded in a CDR.

MPM can be received over outgoing ISUP trunks. Data contained in them must be analyzed and stored in a CDR. These messages can also be transmitted back over the incoming ISUP trunk.

Charging tariff data can be received from an SCP during a call. This data overrides the data provisioned in the MGC charge tables.
The Charge/No-Charge indicator in the ISUP BCI parameter of the ACM/CPG/ANM messages sent to the network by the MGC must be set appropriately based on either provisioned data in the MGC or data received from the SCP.

- Additional INAP requirements

The Cisco PGW 2200 Softswitch can be used to generate Metering Pulse Messages as a basis for charging.

**Note**

Metering information is checkpointed from the active to standby Cisco PGW 2200 Softswitch system every 15 minutes.

The charge table, shown in Figure 1-16, can be accessed using three keys:

- charge origin
- charge destination
- day of the week

The charge table contains the tariff descriptors that are to be applied. The resultant tariff descriptor is in string format and may contain a single tariff id to be applied for the entire day or a list of different tariffs and the time at which they are applied (delimited by spaces).

If the resultant tariff descriptor is a list of different tariffs and the time at which they are applied, the initial entry is a tariff rate to be applied from 0000 hours until the next specified time period, at which point the tariff id following the switch time is applied. A maximum of 5 tariff changes is allowed for a given day, for example, a day may contain 6 different tariff rates.

A tariff descriptor time period value of 0000 indicates the end of time dependent tariff data and the previous (last) found tariff id continues until midnight.

The charge origin may be defaulted (0) when the charging tariff rates are not origin dependent. The day of the week may be defaulted (0) by the craft when the same tariff rate is to be applied to more than one day of the week.

The holiday table allows you to select specific days of the year to be charged differently from the actual day of the week that a holiday occurs on.
In the sample Charge Table (shown in Figure 1-16), the origin/dest charge combination has three entries: ‘1,1,0’, ‘1,1,8’, and ‘1,1,5’. The entry 1,1,8 defines a holiday tariff and 1,1,5 defines a split day tariff for day 5 (Friday). The default entry, ‘1,1,0’, defines the tariff to be applied for all other days (Monday through Wednesday, Saturday, Sunday, and the remaining holiday days 9 and 10).

The split day tariff (see the Charge Table) is interpreted as follows:

- Apply tariff 1 from 0000 - 0700 hours
- Apply tariff 5 from 0700 - 1000 hours
- Apply tariff 1 from 1000 - 1600 hours
- Apply tariff 5 from 1600 - 1800 hours
- Apply tariff 1 from 1800 - 2400 hours
**CLI Charge Origin Table**

The CLI charge origin table is accessed during analysis. It is referenced after a number digit tree analysis when AOC is enabled against the incoming trunk group/sigpath. Valid CLI charge origin table values are 1–9999.

<table>
<thead>
<tr>
<th>CLI Key</th>
<th>Charge Origin</th>
</tr>
</thead>
<tbody>
<tr>
<td>02087568791</td>
<td>2</td>
</tr>
<tr>
<td>01711234567</td>
<td>2</td>
</tr>
<tr>
<td>0403123456</td>
<td>3</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

**Metering Pulse Tariff Table**

The Meter Pulse Tariff Table is indexed using the tariff identifier retrieved from the charge table. The tariff table supports a minimum of 512 (values from 0 to 511) distinct tariffs with user-definable tariff identifiers. Table 1-14 lists the Meter Pulse Tariff Table fields and descriptors. Table 1-15 is a sample Tariff Table example.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tariff Identifier</td>
<td>Independently definable integer.</td>
</tr>
<tr>
<td>Number of Pulses on Answer</td>
<td>Valid values are 0 through 15; a value of 0 indicates that no pulses are generated on receipt of the answer signal.</td>
</tr>
<tr>
<td>Timing Interval Between</td>
<td>Valid values are 500 through 3 600 000 (milliseconds). The minimum interval between consecutive MPMs is 0.5 seconds. A value of 0 indicates that no periodic charge is applied.</td>
</tr>
<tr>
<td>Periodic Pulses</td>
<td>Indicates how many pulses should be sent when the timing interval period expires. Valid values are 0 through 255.</td>
</tr>
<tr>
<td>Periodic Charge Application</td>
<td>At timer expiration, the associated pulses are sent and then the normal periodic interval timer is initiated. Valid values are 0 (synchronous) and 1 (asynchronous). The synchronous method applies the timing interval provisioned immediately upon answer and repeatedly thereafter. The associated meter pulses are transmitted at the end of each timing interval. The isochronous method (also referred to as Karlsson) starts at the first timing interval at a random value ( r ), where ( r ) is in the range of 0-( t ), where ( t ) is the associated timing interval.</td>
</tr>
</tbody>
</table>
You can create a metering pulse tariff table. The MML provisioning commands (for example, prov-add and prov-rtrv) are used to access this table. Each row is referenced by a tariff id that call processing obtains from the charge table. The retrieved row entry contains the tariff rate followed and the scale factor.

**Metering Pulse/AOC Activation**

The metering pulse (and AOC) functionality is controlled by the AOCEnabled property in the properties.dat file (1—enabled, 0–disabled).
Table 1-16 provides information on the data fields used for metering pulse AOC implementation.

### Table 1-16 Charging Parameter Field Definitions

<table>
<thead>
<tr>
<th>Name</th>
<th>Use</th>
<th>Type</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Charge Origin</td>
<td>Trunk group/Sig Path property A-number digit tree result</td>
<td>Integer</td>
<td>1 through 999</td>
</tr>
<tr>
<td></td>
<td>CLI charge origin table</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Charge table</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Charge Destination</td>
<td>B-number digit tree result</td>
<td>Integer</td>
<td>1 through 9999</td>
</tr>
<tr>
<td></td>
<td>Charge table</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Date</td>
<td>Holiday table</td>
<td>String</td>
<td>yy.mm.dd where: yy=00 through 99 mm=01 through 12 dd=01 through 31</td>
</tr>
<tr>
<td>Holiday value</td>
<td>Holiday table</td>
<td>Integer</td>
<td>8 through 10</td>
</tr>
<tr>
<td>Day of week</td>
<td>Charge table</td>
<td>Integer</td>
<td>1 through 10</td>
</tr>
<tr>
<td></td>
<td>Holiday table</td>
<td></td>
<td>Values must be entered as MONDAY–SUNDAY(1–7), HOL1(8), HOL2(9), and HOL3(10)</td>
</tr>
<tr>
<td>Tariff Descriptor</td>
<td>Charge table</td>
<td>String</td>
<td>&lt;tariff descriptor&gt;::= &lt;tariff id&gt; [&lt;&quot; “&gt;] &lt;tariff time switch list&gt;</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tariff Id</td>
<td>Tariff table</td>
<td>Integer</td>
<td>1 through 9999</td>
</tr>
<tr>
<td>Tariff Rate</td>
<td>Tariff table</td>
<td>Integer</td>
<td>1 through 999999</td>
</tr>
<tr>
<td>Scale Factor</td>
<td>Tariff table</td>
<td>Integer</td>
<td>Always set this value to 1 for metering pulses.</td>
</tr>
</tbody>
</table>

### Adding or Removing Country Code

Existing trunk group properties provide national and international prefix digits (such as 0 and 00), which can be used to prepend a national or international format number. However, there is also a requirement to route calls for a given destination to different carriers, which may require only a national format, and does not include a country code. In some cases, carriers may require an international format that includes a country code. When switching between the national and international formats, it is necessary to enhance the existing properties by adding the capability to selectively add or remove the country code.

**Note**

A slight performance impact can be expected if using this function, but it should typically not exceed 5% impact on call processing if the properties described here are provisioned for use.
National Switching Node Operation

As stated, trunk group properties are already present to convert from both national and international formats to the NOA Unknown format. However, it is also necessary to be able to change between the national and international formats by selectively adding the country code, depending on what the country code is, and changing the Nature of Address (NOA) code on a per trunk group (TG) basis.

Figure 1-17  Operation of a National Switching Node

As shown in Figure 1-17, an incoming call for a mobile number arrives from the originating carrier in national format, and the incoming dial plan points to the national dial plan. The national dial plan gives a route result of RouteList #55.

The first route in RouteList #55 contains three trunk groups, each routed to a different carrier:

- **TG1 - Carrier 1**—Is the first choice for mobile calls because it is the least expensive of the three carriers. Carrier 1 accepts national calls only in the national number format; however, it also accepts calls to other countries in the international number format.
- **TG2 - Carrier 2**—Is the second choice because it is less expensive than Carrier 3, but it accepts calls only in the international number format.
- **TG3 - Carrier 3**—Is the last choice because it is the most expensive of the three carriers, and it accepts calls only to international numbers in the Unknown format.

From the previous example, it can be seen that the following items are needed to solve this problem:

- **CC_DIG**—Is the result type used in B-number analysis to record the destination country code for the call. A digmodstring is created so it can be connected to the result in dataword1. See the “Result Type Definitions” section on page 1-17 for additional information on this result.

  For detailed information on provisioning the country code addition or removal capability, see the “Provisioning the CC_DIG Result Type” section on page 4-58.

- **BDigitCCPrefix**—A trunk group-based property which, if enabled, prepends the destination Country code for the call to the B-number (called number) and changes the incoming NOA code to international.
- **CCOrigin**—An incoming trunk group property to record the originating country code for the call.
- **ADigitCCPrefix**—Another trunk group-based property which, if enabled and the NOA code is set to national, prepends the country code from CCOrigin to the A-number (calling number) and changes the incoming NOA code to international.
<Note>
If there is no CCOrigin value present when required, then the A-number is left unchanged.
</Note>

The existing properties to insert digits (A/BnationalPrefix, A/BinternationalPrefix) should also be available, and they are checked after the above properties so they can prepend additional digits, such as adding "00" to the front of the country code and changing the NOA code to Unknown format.

For additional information on trunk group properties, see the Cisco PGW 2200 Softswitch Release 9.8 Provisioning Guide.

When the Cisco PGW 2200 Softswitch is processing a B-number (called number), digits can be added to the front of the B-number. When more than one prefix is added to the B-number, digits are prefixed in the following order:

- **CC_DIG** – The country code to be prefixed to the B-number, if the BDigitCCPrefix property value is set to 1 (enabled)
- **BInternationalPrefix (00)** or **BNationalPrefix (0)**
- **BTechPrefix**

For example, a UK-style national number with a national prefix and a BTechPrefix is: 901444234567 (where BTechPrefix=9, BNationalPrefix=0, and the national number=1444234567). Similarly, the same UK-style national number fully prefixed with country code, international prefix, and a BTechPrefix is: 900441444234567 (where BTechPrefix=9, BInternationalPrefix=00, country code=44, and the national number=1444234567).

**Figure 1-18** is an example showing how these capabilities are used to handle B-number (called number) formats.

**Figure 1-18  Operation of a National Switching Node with Country Code Addition Capability**

As shown in **Figure 1-18**, in the national dialplan the result type **CC_DIG** is set for the national mobile number (CC_DIG=34), so for the calls being routed to each carrier the following occurs:

- **TG1 - Carrier 1** — Is the first choice for national mobile calls because it is the least expensive of the three carriers. Carrier 1 accepts national calls only in the national number format. In this instance the number format is national and needs no modification.
- **TG2 - Carrier 2** — Is the second choice because it is less expensive than Carrier 3, but it accepts calls only in the international number format. To use Carrier 2 the **BdigitCCprefix** property is enabled, the result for **CC_DIG** is prepended to the B-number (called number), and the incoming NOA code is changed to international format.
• **TG3 - Carrier 3**—Is the last choice because it is the most expensive carrier, and it accepts calls only to international numbers in the Unknown format. To use Carrier 3 the `BdigitCCprefix` property is enabled and performed first, changing the number to international format as was done with TG2.

The `BinternationalPrefix` property is also enabled and is performed next. It takes the resulting number (+34612345678) and detects the international NOA code, so it prepends “00” to the number and sets the NOA code to Unknown.

---

**Note**

Calls routed to Carrier 2 (TG2) from the international dial plan do not have `CC_DIG` set, so no modification occurs when international calls get routed to this carrier; they are already in International format with the country code. Calls routed to Carrier 3 (TG3) from the international dialplan are still modified to Unknown format.

---

Figure 1-19 is an example of these capabilities to handle A-number (calling number) formats.

**Figure 1-19  Operation of a National Switching Node with A-Number Formats**

As shown in Figure 1-19, the `CCOrigin` trunk group property is “34” as set in the properties.dat file for the incoming Trunk Group. For calls being routed to each of the three carriers the following occurs:

- **TG1 - Carrier 1**—Is the first choice for national mobile calls because it is the least expensive of the three carriers. Carrier 1 accepts national calls only in the national number format. In this instance the number format is national and needs no modification.

- **TG2 - Carrier 2**—Is the second choice because it is less expensive than Carrier 3, but it only accepts calls in the international number format. To use Carrier 2 the `AdigitCCprefix` property is enabled, and so the value of `CCOrigin` for the call (34) is prepended to the A-number (calling number), and the incoming NOA code is changed to international format.

- **TG3 - Carrier 3**—Is the last choice because it is the most expensive, and it accepts calls only to international numbers in the Unknown format. To use Carrier 3 the `AdigitCCprefix` property is enabled and performed first, changing the number to international format as was done with TG2.

The `AinternationalPrefix` property is also enabled and is performed next. It takes the resulting number (+34612345678) and detects the international NOA code, so it prepends “00” to the number and sets the NOA code to Unknown.
Note

Calls exiting down trunk groups that already have their A-number in international format are unchanged, regardless of the state of the AdigCCprefix trunk group property. This applies to international transit calls where the A-number has already been normalized into international format.

International Switching Node Operation

The requirements for an international switching function are slightly different from those for national switching. In an international switching node, all numbers are normalized into international format (typically the A- and B-numbers) and numbers are switched primarily according to analysis of the country code and area code, such as a country’s mobile code.

However, many carriers (such as PTTs) require numbers routed to destinations within their country to be presented in their national format with the country code removed, and numbers routed to destinations in another country to be presented in international format with the country code still intact.

To provide this capability, the trunk group property BDigitCCrm is required to selectively remove the country code from the B-number (called number) on a per trunk group basis.

If the BDigitCCrm property is set to a non-null value (a country code) and the NOA code is set to international, the initial digits are removed from the B-number if they match the value of BDigitCCrm (that is, if the B-number contains the same country code as BDigitCCrm). The NOA code is also reset to national.

Figure 1-20 illustrates the operation of the BDigitCCrm property in an international switching node.

Figure 1-20 International Switching Node Operation
In the incoming trunk group dial plans, the country code is prepended for incoming numbers that are presented in national format to normalize them into international format. Other modifications can also be made; for example, inserting the country code in the front of the A-number (calling number) for that trunk group and changing its NOA code to international.

The generic international dial plan determines the destination route lists for calls. For calls to national mobile numbers beginning with “61,” the routing priority is Carrier 1, Carrier 2, and Carrier 3. The international dialplan selects RouteList #55 for numbers beginning with “3461,” a number range that is owned by Carrier 1. For each trunk group in Routing List #55 the following treatment is given:

- **TG1 - Carrier 1**—Is the first choice for national number format calls for their mobile number range. The trunk group property BDigitCCrm is set to 34; therefore, calls with NOA code set to international and prefixed with “34” have the country code deleted and the NOA code set to national. All other international numbers are unaffected.

- **TG2 - Carrier 2**—Is the second choice, but accepts calls only in the international number format. Property BDigitCCrm is set to null and calls are routed without modification from international format.

- **TG3 - Carrier 3**—Is the third choice and it accepts calls only to international numbers in the Unknown format. So property BDigitCCrm is set to null and the BInternationalPrefix property is also enabled and is performed next. It takes the resulting number (+34612345678) and detects the international NOA code, so it prepends “00” to the number and sets the NOA code to Unknown.

For numbers sent to Route List #14, Carrier 4 is the first choice and Carrier 1 is the second choice:

- **TG4 - Carrier 4**—Is the first choice, but this carrier requires the B-number (called number) to be presented in national format. The trunk group property BDigitCCrm is set to “44;” therefore, calls with the NOA code set to international and prefixed with “44” have the country code deleted and the NOA code reset to national. All other international numbers are unaffected.

- **TG1 - Carrier 1**—Is the second choice and only accepts calls for Country code “44” in international format. With property BDigitCCrm equal to “34,” the only called numbers that have their country code prefix removed are those with prefix digits of “34.”

There are two ways of routing calls to country code “44”:

- On TG4 by sending national format (by deleting the country code “44”)
- On TG1 by sending international format (by leaving the country code “44” intact)

**Note**  Setting property BDigitCCrm to 34 has no effect on calls to country code “44.”

### Failing to Find Country Code Digits

Consider what actions you want the Cisco PGW 2200 Softswitch takes if an occurred error can jeopardize call completion, for example, the call fails for any reason. In the unexpected event that a country code is not present when it is required, one of the following events occurs.

- **A-Number handling**—If a country code prefix is directed to be applied by the ADigitCCPrefix property, but the digits are not available from the CCOrigin trunk group property, then the A-number and NOA code are not changed. No action is taken and the call is allowed to continue.
• **B-Number handling**—If the CC_DIG result is expected during processing, but it is not retrieved due to an error (such as a configuration error), one of the following incorrect actions can result:

  - If only the country code prefix is required, it is not applied and the egress IAM contains the numbers in the received national format and the NOA code is national format.
  
  - If both the international number and country code prefix are required, the international prefix is applied without the country code and an incorrect number is sent forward to the next switch.

  If the BDigitCCPrefix trunk group property is set to enable country code addition functionality, but the processing fails due to absence of the required country code digits, the call is forced to fail by setting an IC_TEMPORARY_FAILURE cause and proceeding to Cause analysis.

  To ensure that this occurrence is noted, an internal alarm is raised and an associated log message is issued indicating that a prefix addition has failed.

**Action If Country Code Removal Leaves the B-Number Empty**

When operating in Overlap mode, it is possible that the number of digits received is sufficient to enable onward routing, but is not sufficient to leave digits in the B-number if the country code is removed. This is a case where the country code digits are routed against, but are then removed. This case must be guarded against when processing calls.

Consider the following example:

• B-number as received in an overlap operation is 34, 621, and 345678.

• B-number analysis with only the initial digits (34) present yields a ROUTE result.

• The trunk group property setting for the designated route is BDigitCCrm = 34.

  When operating in Overlap mode and analyzing the B-number, routing is made against the initial set of digits (“34”), then the trunk group property BDigitCCrm requires that these leading digits (34) be removed before sending the IAM to the next switch, leaving no digits in the B-number.

  The terminating call control (TCC) protocol rejects this call because it fails the 0 digits check, but to inform the user of the root cause of the failed call (a configuration error), the call clear down is invoked internally, an alarm (CCCodeModfailed) is generated, and the associated message log indicating “prefix removal failure” is issued.

  This problem can be avoided when configuring an Overlap system by ensuring that any Route result is provisioned after the country code digits, allowing for their potential removal. For example, taking the number used in the previous example:

  • B-number as received in an overlap operation is 34, 621, and 345678.
  
  • B-number analysis with the digits 34621 present yields a ROUTE result.
  
  • The trunk group property setting for the designated route is BDigitCCrm = 34.

  Routing is performed only after receiving overlap digits 34621. After country code removal, the IAM sends forward a B-number of 621. All other digits of the B-number are received and forwarded in subsequent address messages (SAMs).

**TNS Feature**

Within the Pre-analysis area, a stage is supported called TNS analysis. This stage provides an analysis capability against the transit network selection parameter information (or the Carrier Selection parameter information) as received in the incoming message.
As with the other Pre-analysis stages, all of the Pre-analysis results are available to this stage; but specifically the capability to switch the call according to a received carrier ID is supported. The carrier ID is an over-decadic or decadic character string that must provoke specific routing. Examples of carrier IDs are D001, B77, 88, and 23456. The carrier ID is received in the incoming message (in the parameters previously described) and is then extracted to analyze in this Pre-analysis stage.

The following capability is provided if a CarrierID is received in an incoming message:
- Route the call according to the CarrierID string.
- Route the call according to the B-number and ignore the CarrierID.
- Block the call according to the presented CarrierID.

The extracted OrigCarrierID string is used as input to this Pre-analysis stage and the following possible results returned provide the previously listed capabilities.
- ROUTE, COND_ROUTE, or PERC_ROUTE—If any of these result types are returned, then the call is to be routed according to the OrigCarrierID. The resulting data has routing information that is used to immediately start the Routing analysis stage.
- If the call is to be routed according to the B-number and not the CarrierID, then this is indicated by no results being returned from this stage, or just returning a specific result type (for example, AMODDIG) that has Routing or Blacklist results).
- BLACKLIST—If this result type is returned, then the call is to be blocked according to the CarrierID, an internal result is set to reflect this, and provoke call rejection.

To avoid the possibility of a routing loop where the MGC passes the call back to the preceding switch that originated the call, some trunk group property checks are performed at the protocol level.

The property is provisioned against the incoming trunk group where it is expected that TNS information is received. The OrigCarrierID property contains a CarrierID string identifying the previous switch. The incoming protocol reads this property when processing the TNS information and verifies that the CarrierId in the TNS parameter matches the OrigCarrierID parameter provisioned in the trunk group property. If they do match, then a routing loop exists and the MGC rejects the call and sets cause to “Normal Unspecified”. However, if there is no match on OrigCarrierID, then the protocol completes processing and sends the data to analysis where the Pre-analysis stages are the first to be performed.

Routing Analysis

In a call control environment, Routing analysis normally occurs after Number analysis and ultimately provides the means to traverse the routing lists, route, and trunk group data. Additionally, Routing analysis can be invoked by the Pre-analysis stage or the Cause analysis stage. The purpose of Route analysis is to find a trunk group within a set of routes that can be selected to be used to route the call to the desired destination.

Routing analysis is started when Pre-analysis, B-number analysis, Cause analysis, or Conditional Routing analysis returns a Route List Name (see Figure 1-21). The output from the Route List Name is used to access the Route List, from which the search for routes and trunk groups is started.

The purpose of Route analysis is to select a trunk group. Route preferences or bearer preferences present in the incoming IAM or Setup message are read and applied during the route selection process.

Based on customer requirements and preferences, the transmission medium can be selected by the Route analysis function choosing the appropriate medium from the selected trunk group list. Additionally, trunk groups could be selected based on bearer preferences or requirements present in the incoming message. If that is the case, the order of trunk group selection would be influenced (for example, ISUP essential indication).
Note

Although routing analysis and route selection are part of the call routing process, they are not used by the Cisco PGW 2200 Softswitch in signaling control mode. In the signaling control mode, all routes are static, or “nailed-up,” and the outgoing trunk is based on the trunk (or circuit) used by the incoming call.

**Routing Terminology**

The following terms are used when describing Routing analysis.

Route list—A collection of routing alternatives that can be used to transport a call between its origin and its destination. The individual routes comprising the route list provide routing to the same destination, but use different physical paths.
Route—A collection of trunk groups with a common destination and may be listed in more that one route list. For example:

**Table 1-17 Route Example**

<table>
<thead>
<tr>
<th>Route</th>
<th>Containing Trunk Groups</th>
<th>Trunk Group Signaling</th>
</tr>
</thead>
<tbody>
<tr>
<td>Herndon_Sterling</td>
<td>TKGrp 1</td>
<td>ANSI SS7</td>
</tr>
<tr>
<td></td>
<td>TKGrp 2</td>
<td>Q761 ISUP</td>
</tr>
<tr>
<td></td>
<td>TKGrp 3</td>
<td>PRI</td>
</tr>
<tr>
<td></td>
<td>TKGrp 4</td>
<td>IP</td>
</tr>
<tr>
<td></td>
<td>TKGrp 5</td>
<td>SIP</td>
</tr>
</tbody>
</table>

**Note**

All trunk groups are connected between Herndon and Sterling.

Trunk group—A collection of like circuits or channels (for example, all SS7 circuits with echo cancellers connected) that connect the same two end points. All circuits within a trunk group have the same signaling route, (that is, a signaling route is an attribute of a trunk group). A trunk group may be listed in more than one route.

### Routing Analysis Components

As shown in Figure 1-21, Routing analysis has the following components:

- Percentage based routing
- Time conditional routing
- Route list analysis

Percentage based routing and time conditional routing provide a flexible start to Routing analysis and can interact with each other as required. The final output of these two stages provides a starting point into the Route list analysis stage, in which Routing analysis is completed. Each Routing analysis component can be selected individually, and depending on the analysis type, may lead to another analysis stage, as listed in Table 1-18.

**Table 1-18 Routing Analysis Interaction**

<table>
<thead>
<tr>
<th>Present Analysis Stage</th>
<th>Next Analysis Stage Possible</th>
</tr>
</thead>
<tbody>
<tr>
<td>Route list analysis</td>
<td>None (this is the final analysis stage)</td>
</tr>
<tr>
<td>Percentage based routing</td>
<td>Time conditional routing</td>
</tr>
<tr>
<td></td>
<td>Route list analysis</td>
</tr>
<tr>
<td>Time conditional routing</td>
<td>Percentage based routing</td>
</tr>
<tr>
<td></td>
<td>Route list analysis</td>
</tr>
</tbody>
</table>

During Route list analysis, when trunk group data is being read and the trunk group is being assessed for suitability, route preferences or bearer preferences (some present in the received IAM, Setup, or INVITE message and some previously collected from Number analysis) are used in the selection process.
Number Termination

A result type present in the dial plan, called TERM_INFO and configured early in the B-number analysis, indicates whether a full B-number analysis might be required to determine the final destination. On detection of the TERM_INFO result type, the called number is searched in the Number Termination table. The search returns a route list name used to start routing analysis. This avoids the need for a large dial plan and reduces memory consumption when it is loaded.

Percentage Based Routing

The percentage based routing* permits the user to distribute the traffic load across route lists based on assigned percentage values. If percentage based routing is combined with time conditional routing, you can fine tune the routing distribution according to day and time entries. For example, it is possible to divide traffic 60-40 between France and Germany with traffic to Germany routed to Berlin from 9:00 a.m. to 5:00 p.m., to Frankfurt from 5:00 p.m. to 12:00 p.m., and to Munich from 0:00 a.m. to 9:00 a.m. It is also possible for traffic that cannot be progressed on one of the percentage options to optionally overflow to a completely different route, or be re-directed onto one of the other percentage-shared routes.

As shown in Figure 1-22, the percentage based route Italy (% based route Italy) has three routing options available. Route list Rome has a 50% weighting, route list Florence has a 20% weighting, and a 30% weighted route that is switched according to time of the day based on the conditional table, tea, which ultimately leads to route list Venice and route list Turin.

![Figure 1-22 Example of Percentage Based Routing Level Overflow](image_url)

Routing Overflow

Within the percentage based routing functionality, the overflow capability exists that allows you to assign percentage values to the routes if congestion occurs. The overflow methods are:

- If congestion occurs on the selected route list, then overflow to the route list with the next highest percentage value. Continue this until the call is completed or until all routes are exhausted and then set an internal cause value and invoke Cause analysis with the goal of clearing the call.
If congestion occurs on the selected route list, then overflow to the route list with the next highest percentage value. If all routes are exhausted, then overflow occurs to a final overflow route list that is used only when all other options have been attempted and were all unsuccessful.

Handling of Overflow at the Percentage Based Route Level

At the percentage based routing* level there is the capability to choose to overflow to the next route list if all trunks in the previously chosen route list or time of day are unavailable. There is no capability to return to the time of day directly, as this would produce the same route list again. There is a parameter at the percentage based routing level to specify whether overflow is supported. By default overflow will be enabled. If overflow is disabled the call will go to cause analysis with a well-known cause code if no trunks are available in the first route list.

Figure 1-23  Example of Percentage Based Routing Level Overflow

![Diagram showing percentage based routing with overflow]

The example in Figure 1-23 shows that if from the percentage-based route called Italy the route list “Rome”, time of day “Tea” or route list “Florence” can be chosen. If the random algorithm chooses route list ‘Rome” and the trunks were not available it would be possible then to use either the time of day route “Tea” or route list “Florence” if overflow were enabled at the route list level. In the case where the route list required from a PERC_ROUTE result cannot be selected because of congestion then try the other route lists, starting with the highest percent. Therefore time of day “Tea” would be chosen and if all trunks were not available then it would be possible to use route list “Florence”. If all trunks were unavailable in route list “Florence” as well the call will go to cause analysis. From cause analysis the call can be terminated with a well known cause value or further analysis can be supported, if the call enters this cause value twice the call will be released.

*Indicates software Release 9.3(2) functionality.
Handling of Overflow at the Route List level

At the route level if all the trunks in the route are unavailable then there will be overflow into the next route in the route list if one is configured. As shown in Figure 1-25 route list 1 would use route NY and would overflow into route DC if all trunks were unavailable in route NY.

Handling of Overflow at the Route Level

Currently overflow is handled at the trunk group and the route level. At the trunk group level overflow is possible if the user has configured the route to have sequential selection. If all the trunks in a trunk group are unavailable the routing will overflow into the next trunk group in the route, if one is configured. In the example shown in Figure 1-26, if all trunks in trunk group 1 are unavailable then calls using route A would then be overflowed into trunk group 2. Note that if distribution is turned on a random number is used to provide the offset into the route table that is effectively a trunk group list, so overflow is not supported.
When a user creates a percentage-based route, he or she connects the entries to route list names or time of day route names. The first entry created is the primary entry. The primary entry is used as the default routing condition for any load that the user does not explicitly set. The load cannot be modified on the primary entry in the percentage based route table because it is automatically changed when the load is modified for other entries in the table.

The sum of all load values added to the percentage based routing name cannot exceed 100. If the sum of the load values, not including the primary, matches 100, the primary entry has a value of 0.

It is enforced that a conditional route name cannot be connected to the percentage based routing if there is currently in the conditional route table any percentage based routing name connected to the same conditional route name to prevent routing loops. The same route list name or conditional route name cannot be added to the percentage based route name multiple times; this same functionality can be obtained by changing the value of the load. The primary route can be deleted only if the whole percentage based route is deleted, since a percentage based route has to be configured for every percentage based route name.

The overflow set allows you to define if the percentage based route handles overflow. If the overflow set is changed, it is configured for all entries in the percentage based routing name. An overflow can only be associated with a percentage base route name if the overflow set is enabled. The overflow defines a routing condition that is used only if all percentage based routes with a defined load have been exhausted. There can be only one overflow route for each percentage based routing name.

A maximum of five route list names and conditional route names can be configured in a percentage based routing name.

### Time of Day Routing

Time of day routing* lets you select a route list or an entry point into the percentage based routing based on the time of day, and day of week.

*Indicates software Release 9.3(2) functionality.

---

**Note**

Some performance impact can be expected when the time conditional routing function is invoked; however, normally it should not exceed 5% on call processing.
Conditional Route Description

When a user creates a conditional route description, he or she connects the entries to route list names or percentage based routing names. The first entry created in a conditional route description is the primary entry. The primary entry is used as the default routing condition for any time period that the user does not explicitly set. As a result, the primary entry does not have a time period associated with it. Therefore, the primary route can only be deleted if the whole conditional route description is deleted, because a conditional route has to be configured for every conditional route description name.

The supported time periods are from 0000 to 2359, where the times can be configured in 15-minute increments. Time periods cannot overlap currently existing start and end times. For example, if the time period 1000 to 1200 is configured, then 0900 to 1100 and 1130 to 1300 cannot be configured; however, 1000 to 1200, 0900 to 1000, 1200 to 1300, and 1030 to 1200 can be configured.

To prevent routing loops, a percentage based routing name cannot be connected to the conditional route name if there is currently in the percentage based routing table any conditional route name connected to the same percentage based routing name. The conditional route description name cannot be deleted if any conditional route is connected to it.

A maximum of five route list names and percentage based routing names can be configured in a conditional route description.

Conditional Route

Conditional routes are connected to conditional route description names based on the day of week. The first entry created in the conditional route is always the default day of week. It is used to provide a default routing condition for any day of week that the user does not explicitly set. The default day of week can be deleted only if the whole conditional route is deleted. Each conditional route supports a default entry, seven days, and three holiday entries.

Route Holiday

The route holiday allows dates to be specified with the three holiday days. When a call is received that is destined for conditional routing, the holiday days are used instead of the default entry; or the days, Monday through Sunday, are used if a holiday day is associated with the actual date.

Provisioning route holiday allows you to separately enter holidays for routing purposes. The route holiday list contains all the valid holidays for a given user.

<table>
<thead>
<tr>
<th>Date</th>
<th>Holiday Day</th>
</tr>
</thead>
<tbody>
<tr>
<td>2001.11.22</td>
<td>HOL1</td>
</tr>
<tr>
<td>2001.11.23</td>
<td>HOL2</td>
</tr>
<tr>
<td>2001.12.09</td>
<td>HOL3</td>
</tr>
<tr>
<td>2001.12.10</td>
<td>HOL3</td>
</tr>
<tr>
<td>2001.12.24</td>
<td>HOL2</td>
</tr>
<tr>
<td>2001.12.25</td>
<td>HOL1</td>
</tr>
<tr>
<td>2001.12.31</td>
<td>HOL2</td>
</tr>
<tr>
<td>2002.01.01</td>
<td>HOL1</td>
</tr>
</tbody>
</table>
The route holiday consists of the date and the holiday day.

- **Date**—The date is entered in the format yyyy.mm.dd. The following ranges apply:
  - yyyy = 2000-9999
  - mm = 01-12
  - dd = 01-31
- **Holiday Day**—This number is a positive integer that indicates the holiday day. The valid values are:
  - Hol1
  - Hol2
  - Hol3

## Route List, Route, and Trunk Group Data Overview

### Route List

The route list consists of a sequentially selected list of routes with a distribution entry against each route. If distribution is enabled, a random number is used to provide the offset into the route table, which is effectively a trunk group list. If distribution is disabled, then the routes are chosen in sequential selection.

A route list entry can be entered from:
- The percentage route.
- The time of day routing.
- The number termination.
- The dial plan from a ROUTE or ANNOUNCEMENT result type.

The system gives a warning if more than 20 routes are created in the route list since only the first 20 routes are used.

### Routes

Routes represent the sequential selection used to choose the trunk groups in the route. When distribution is enabled in the route list, you can enable the weighted trunk groups feature so the same trunk groups can occur in the same route multiple times.
Routing Trunk Groups

Screening is performed upon the trunk group list based on the selection adjuster. If an essential value is present (in the incoming message), only trunk groups matching this criterion are kept and the other trunk groups are discarded. If no entry matches the essential criteria, then trunk group data is retrieved for the next route in the route list. Again, the selection criterion is applied and this process continues until a trunk group is found that matches the selection criterion. If no trunk group can be found that matches the essential criteria, a routing failure is declared and the call is released with a cause message. If an excluded value is present, then any trunk group matching this condition is discarded. If a preferred indication is present, then two separate lists are created (primary and secondary) and trunk groups matching the preferred criteria are placed in the first list (primary), which is initially used to select an appropriate outgoing circuit or path.

If TDM trunk groups are present in the screened list, and bearer preferences exist, then these preferences are used to select the appropriate trunk group using a procedure similar to the one referenced above. If no bearer preference is given, then the first trunk group is chosen. Analysis returns the selected trunk group (if one is available).

When a trunk group is selected, it is removed from the route list so it cannot be re-selected if a new trunk group ID (circuit selection having been unsuccessful on the existing one) is requested.

Similarly, if a route has been selected, it is also removed from the route list to avoid re-selecting it later. The analysis function retains the route list data and trunk group data, as appropriate, so it is ready for any further trunk group requests regarding this call instance.

If an indication is received for congestion or unavailability from circuit selection, then analysis is reinitiated to provide another trunk group or route from the lists in analysis data.

TDM Trunk Group Attributes

TDM trunk group attributes include cut through, queue timer, and reattempt number.

Queuing

You can provision a queuing capability (by provisioning a value in TDM Attributes against TDM trunk groups). If circuit selection on a trunk group has failed (with the response of trunk group congestion), the MGC waits for a circuit to become free for the duration of this value returned from the TDM Attributes. Thus the call is queued.

A queue timer value is returned from analysis (read from the TDM Attributes), where queuing is applicable (if so provisioned). If a value other than 0 is returned, this value is used if a circuit is unavailable. The call is then placed in the FIFO queue associated with that trunk group, to wait for an available circuit.

Once a congestion message is received, the queue timer (set to the value received from analysis) holds the call while waiting to receive an indication that a circuit is available.

When a circuit is available, the first call queued is removed from the list, assigns the circuit to it, and indicates normal call processing can continue. However, if no circuit is selected and the queue timer expires, then the call is released with a congestion indication.

If a forward release is received during queuing, the call is removed from the list when the call is released.
Repeat Attempts (Re-Attempts)

It is possible to make a number of repeat attempts to select a circuit on the same trunk group. The maximum number of attempts is provisioned in the TDM Attributes by the ReAttempts field and is delivered from analysis. If a value other than 0 is provisioned, it indicates that under certain conditions reattempts are made to select a circuit on the current trunk group.

If reattempts are not provisioned (default value is 0), then no reattempts are made. Thus, if a circuit-selection request is made and the response message is TrunkGroupCongestion, analysis is resumed for a new trunk group that is not congested. In this case reattempts are not made. If there are no more trunk groups available, a cause is set and the call is released.

Circumstances in which a reattempt is made on the same trunk group are:

- If glare occurs when setting up the TCC side.
- If COT failure occurs when setting up the TCC side.
- If when the outgoing IAM is sent forward (on TCC side) it is immediately responded to with a release message from the subsequent exchange containing a cause reflecting one of the following internal cause values:
  - IC_CH_UNACCEPTABLE
  - IC_NO_CIRCUIT_AVAILABLE
  - IC_NETWORK_OUT_OF_ORDER
  - IC_ACCESS_INFO_DISCARDED
  - IC_SWITCHING_EQUIP_CONGESTION
  - IC_REQ_CIRCUIT_UNAVAIL
  - IC_RESOURCES_UNAVAIL_UNSPEC
  - IC_TEMPORARY_FAILURE
  - IC_CHANNEL_OUT_OF_SERVICE
  - IC_PRIORITY_FORCED_RELEASE
  - IC_PRECEDENCE_BLOCKED
  - IC_PREEMPTION
  - IC_PROTOCOL_ERROR_UNSPEC
  - IC_OPERATOR_PRIORITY_ACCESS
  - IC_REPEAT_ATTEMPT

Where reattempt is enabled, actions depend on the reattempt value. If this value is greater than 0, the TCC side is disconnected and destroyed, call context is restored to the pre-circuit selection status, the Reattempts counter decrements by 1, and then circuit selection starts again on the same trunk group. If the same response occurs again, this process repeats, each time decrementing the counter.

Once the reattempt counter is 0, at the next occurrence, the TCC is disconnected and destroyed, call context is restored as before and goes back to generic analysis for a new trunk group upon which to attempt circuit selection. If generic analysis finds that no circuits are available but there are further routes it will handle this, select the available trunk groups and eventually return a new trunk group. If there are no more routes or trunk groups it returns indication of No More Trunk Groups and a corresponding cause value.
SIP Trunk Group Attributes

SIP trunk group attributes include: SIP URL, port number, SIP version, cut through, extension support, service resource record, and bearer capability name. These attributes are used to configure the SIP routing trunk group.

Routing Features

Weighted Trunk Groups

Weighted trunk group based routing was implemented by allowing the same trunk group to be used multiple times in the same route when the random distribution algorithm is enabled. The user has the option of setting the distribution indicator on the route entry to determine how the trunk groups are selected in each route. If the distribution is OFF then sequential selection is used to choose the trunk groups in that route. If the distribution is ON then random selection is used. This is useful if there is a need to balance the choice of trunk groups across the connected equipment. If weighted trunk group based routing is enabled or disabled at the route level then the following rules must be maintained:

- If weighted trunk group based routing is enabled at the route any route list that it is connected to must have distribution enabled to ensure that the random algorithm is used.
- If weighted trunk group based routing is enabled at the route then the same trunk group can be added to the route multiple times.
- If weighted trunk group based routing is disabled at the route then the same trunk group cannot be added to the route multiple times.
- If the route has the same trunk group connected to it more than once the weighted trunk group based routing cannot be changed to disabled.
- If the route list is connected to a route that has weighted trunk group based routing enabled, the parameter distribution on the route list cannot be changed from on to off.
- If the route has weighted trunk group based routing as enabled, the user cannot create a next trunk group.
- When the user deletes the trunk group from a route that has multiple trunk groups of the same value, he or she will delete the first one in the list only. The action is successful. A warning shows that there is still x instances of the trunk group configured in the route.
- The number of trunk groups in a route is limited to 100.

Carrier Translation

To support NNP, to determine route selection is made for a particular carrier. When a call is received with the Transit Network Selection parameter containing a CarrierID. If the XECfgParm.dat property for VSCNetworkPlacement indicates “Nanp_AT”, then selection is made when searching the Route List to only choose route lists supporting that connection to that particular carrier. In the Route List, the Carrier ID field allows this cross referencing during route selection.

This functionality is removed and replaced by TNS in software Release 9.3(2).
Trunk Group Preferences

The trunk group preference can be chosen from the dial plan using the ROUTE_PREFERENCE result type, the Bearer Preferences set in the Forward call indicator (FCI) in the IAM or Setup message, or the RoutePref property. This is indicated by preferences carried in the SETUP/IAM message received on the incoming side and comprises values such as ISUP essential, ISUP preferred etc. This selection is applied to fine tune the trunk group choice to provide the most suitable type of trunk group for this call.

When route analysis encounters a selection method of sequential assigned to a route (distribution field), all the associated trunk groups contained within the route are retrieved by reading the relevant route block data. Screening is performed on this trunk group list based upon the selection adjuster. If an essential value is present, only those trunk groups matching this criterion are kept and all other trunk groups are discarded. If no entry matches the essential criterion, then trunk group data is retrieved for the next route in the route list. If this route has a selection method of sequential, then the selection criteria are again applied and this process continues until an appropriate trunk group is found that matches the selection criteria.

If no trunk group is found that matches the essential criteria, then a routing failure indication is declared. If an excluded value is present, then any trunk group matching this condition is discarded. If a preferred indication is present, then two separate lists (primary and secondary) are created and trunk groups matching the preferred criteria are placed in the first list which is initially used to select an appropriate outgoing circuit or path.

If TDM trunk groups are present in the screened list and bearer preferences exist, then this is used to select the appropriate trunk group using a procedure similar to the one described before. If no bearer preference is given, then the first trunk group is chosen. Analysis returns the selected trunk group (if one is available).

When a trunk group has been selected, it is removed from the list so that it cannot be re-selected if another search is performed to select a new trunk group ID (that is, circuit selection has been unsuccessful on the existing trunk group). Also, if a route has been selected, it is also removed from the element list to avoid re-selecting it later. The analysis function retains the route list data and trunk group data as appropriate ready for any more trunk group requests regarding this call instance. If a route has been selected, it is also removed from the element list to avoid re-selecting it later.

If a congestion or unavailability indication is returned from circuit selection, then analysis is recalled to provide another trunk group or route from the lists retained in analysis data.

Bearer Capability Based Routing

Bearer capability based routing is used during route analysis. The route selection process first checks the call bearer capability against the trunk group list. If a match is found, then route selection moves to the next trunk group. If every trunk group of every route does not allow TMR, then the call is released with an internal cause code (RELEASE_NO_BEARERCAP_NOT_POSSIBLE). This cause code is mapped to Q.850 cause code 57 (Bearer capability not authorized).

Configure a bearer capability blacklist for each trunk group using the internal TMR values to configure this list.
Codec Selection

The GWDefaultCodecString property is provisioned against a trunk group, referred as level 2 codec configuration and against the MGCP sigpath, referred to as level 1 codec configuration. The level 2 property is read if there is no level 3 (dial plan) configuration or if the level 3 configuration for the codec list or the codec is defined as preferred.

The level 1 property is read if the level 2 codec configuration is not present and the level 3 configuration is defined as preferred or there is no level 3 configuration. A default value for the GWDefaultCodecString property is used if neither level 2, nor level 1, nor level 3 codec configuration is present. The resulting codec list from level 3, level 2, or level 1 is then sent to the incoming gateway in the Local Connection Option parameter of the create connect (CRCX) message.

If none of the codec levels are configured default level 0, which is “NULL” is used. When this occurs, the MGC does not participate in codec negotiation other than to deliver the message by the MGCP gateway. The MGCP gateway supports codec negotiation by the transfer of preferred codecs in the SDP messages exchanged between the ingress and egress gateways. The decision for the codec to be used is made at the gateway. Currently the ingress gateway sends a list of codecs in the SDP response to the egress gateway. If any codec configuration levels are configured, they override level 0.

Note

The egress gateway determines which codec is used.

The Cisco PGW 2200 Softswitch supports different codec level configurations to influence the codec negotiation by providing the configured codec list in the local connection option (LCO), which limits the list of codecs from which the gateway can select. The gateway always responds with one of the codecs from the list.

Note

On the ingress side, for level 3, if preference is mandatory, it overrides level 2 and level 2 overrides level 1 and level 1 overrides level 0. However, on the egress gateway if level 2 is configured level2 overrides all other levels.

Level 3 codec configuration allows the CODEC result type to be set in A number or B number analysis. If the result specifies the preference, in dataword2, as mandatory, then the codec list from previous levels is ignored and the egress call supports the codec specified in dataword1. When dataword2 is configured to be preferred, then the codec list from the previous applicable level is appended after the codec specified in dataword1.

Note

Level 3 codec configuration overrides previous codec levels if the preference is mandatory on the ingress side.

Route Advance

Cisco PGW 2200 Softswitch retrieves a trunk group no knowing the route in which this particular trunk group resides. During this process, if all the trunk groups on a particular route have been exhausted, then the next route in the route list is selected and the search for a suitable trunk group continues until one is found or the route list is exhausted and then the search begins again.