NIS D354-1

UCS DMS-250

NetworkBuilder Carrier AIN

TCAP Protocol Definition

UCS15 (CSP15) Release 08.01 June 2001

NORTEL NETWORKS

NOTICE

This specification is provided as a guide for network planners and suppliers of systems and equipment designed to meet the requirements of Nortel Network's DMS-250 Carrier AIN TCAP specification. Nortel Networks reserves the right to revise the contents of the specification for any reason, including, but not limited to, conformity with standards promulgated by any public standards agency, advances in technology, or to reflect changes in requirements of communication networks, systems, or applications.

Nortel Networks makes no representation with respect to and does not warrant the technical information contained in this specification including, without limitation, any warranty as to fitness for a particular purpose or merchantability, or against infringement of patents, copyrights, trademarks, or other intellectual property rights.

The provision of any capabilities described in this document is dependent on certain business decisions, resolution of which may result in changes to, withdrawal of, or addition to, any of the capabilities described in this document.

Receipt and use of the technical information contained in this specification by the recipient shall constitute agreement by the recipient that it waives any claims it might have against Nortel Networks as a consequence thereof and that in no event shall Nortel Networks be liable for any special, indirect, incidental, or consequential damages of any nature whatsoever.

UCS DMS-250

NetworkBuilder Carrier AIN

TCAP Protocol Definition

Publication number: NIS D354-1
Product release: UCS15 (CSP15)
Document release: Release 08.01
Date: June 2001

PROPRIETARY
Released under license

Printed in the United States of America

NORTEL NETWORKS PROPRIETARY: The information contained in this document is the property of Nortel Networks. Except as specifically authorized in writing by Nortel Networks, the holder of this document shall keep the information contained herein confidential and shall protect same in whole or in part from disclosure and dissemination to third parties and use same for evaluation, operation, and maintenance purposes only.

Information is subject to change without notice. Nortel Networks reserves the right to make changes in design or components as progress in engineering and manufacturing may warrant

DMS, MAP, NetworkBuilder, Nortel, and SuperNode are trademarks of Nortel Networks.

Publication history

January 2001

Preliminary 08.01 released for UCS15 (CSP15)

May 2000

Preliminary 07.01 released for UCS14 (CSP14)

January 2000

Standard with changes for UCS11 (CSP11)

March 1999

Preliminary 06.01 released for UCS11 (CSP11)

January 1999

Preliminary 05.02 released for UCS09 (CSP09)

September 1998

Preliminary 05.01 released for UCS09 (CSP09)

April 1998

Standard 04.02 released for UCS08 (CSP08)

December 1997

Preliminary 04.01 released for UCS08 (CSP08)

October 1997

Standard 03.02 release for UCS07 (CSP07)

April 1997

Preliminary 03.01 release for UCS07 (CSP07)

February 1997

Re-released as NIS D354-1 as Standard 02.03 for UCS06 (CSP05)

January 1997

Standard 02.02 release for UCS06 (CSP05)

October 1996

Preliminary 02.01 release for UCS06 (CSP05)

May 1996

Preliminary 01.01 release for UCS05 (CSP04)

Contents

Publication history v
January 2001 v
May 2000 v
January 2000 v
March 1999 v
January 1999 v
September 1998 v
April 1998 v
December 1997 v
October 1997 v
April 1997 v
February 1997 v
January 1997 vi
October 1996 vi
May 1996 vi

About this document xv
Introduction xv
Intended audience xv
Applicability of this document xv
How this document is organized xv
How to check the version and issue of this document xvi
References in this document xvi
How requirements are presented xvii
Absolute labels xviii
Local labels xviii
How AIN identifiers are presented xviii

Chapter 1: Introduction 1
1.1 AIN overview 1
1.2 Implementation phases 2

Chapter 2: Messaging overview 3
2.1 Introduction 3
2.2 Message types 3
2.2.1 Call-related messages 3
2.2.2 Non-call related messages 3
2.2.3 Error messages 4
2.3 Exclusions and limitations 4

2.3.1 Message transport 4
2.3.2 Message set restrictions 4
2.3.3 Parameter set restrictions 9
2.3.4 Message responses 12
2.4 Parameter mapping 13
2.5 Extensions to AIN TCAP 22
2.5.1 Originating Feature Requested trigger 22
2.5.2 AINDigits parameters 22
2.5.3 Other AIN parameters 23
2.5.4 Extension parameters 23
2.6 Office Code trigger 24
2.7 Protocol version control 24

Chapter 3: SSP call-related messages 29
3.1 Introduction 29
3.2 General requirements 29
3.2.1 Messages 29
3.2.2 UserID parameter 30
3.2.3 Billing numbers 30
3.2.4 Calling party identification 31
3.3 Close message 32
3.4 Info_Analyzed message 33
3.5 Info_Collected message 35
3.6 Network_Busy message 37
3.7 Origination_Attempt message 40
3.8 O_Answer message 42
3.9 O_Called_Party_Busy message 42
3.10 O_Feature_Requested message 45
3.11 O_No_Answer message 47
3.12 O_Term_Seized message 49
3.13 O_Mid_Call message 50
3.14 Termination_Attempt message 53
3.15 O_Disconnect message 55
3.16 Timeout message 56
3.17 Call_Info_From_Resource message 56
3.18 Resource_Clear message 57
3.18.1 Intelligent Peripheral Requirements 59
3.19 CTR_Clear message 59
3.19.1 Intelligent Peripheral Requirements 60
3.20 O_Abandon message 61
3.21 Failure_Outcome message 62

Chapter 4: SCP call-related messages 65
4.1 Introduction 65
4.2 General requirements 65
4.2.1 Routing 66
4.2.2 Outpulsing 66
4.2.3 Interworking with PSA 67
4.3 Analyze_Route message 67
4.4 Cancel_Resource_Event message 70

4.5 Continue message 71
4.6 Disconnect message 72
4.7 Send_To_Resource message 73
4.7.1 Intelligent Peripheral Requirements 76
4.8 Authorize_Termination 76
4.9 Connect_To_Resource 78
4.9.1 Intelligent Peripheral Requirements 80
4.10 Collect_Information 80
4.11 Call_Info_To_Resource 81
4.12 Originate_Call message 82
4.13 Merge_Call message 85
4.14 Disconnect_Leg message 85
4.15 Acknowledge message 86

Chapter 5: Non-call Related Messages 89
5.1 Introduction 89
5.2 General requirements 89
5.3 SCP non-call related messages 90
5.3.1 Request_Report_BCM_Event message 90
5.3.2 Send_Notification message 91
5.3.3 ACG message 91
5.3.4 ACG_Global_Ctrl_Restore message 93
5.3.5 Furnish_AMA_Information message 93
5.4 SSP non-call related messages 94
5.4.1 Termination_Notification message 94
5.4.2 ACG_Overflow message 96
5.4.3 ACG_Global_Ctrl_Restore_Success message 98

Chapter 6: Error messages 99
6.1 Introduction 99
6.2 General requirements 99
6.3 Application_Error message 101
6.4 Close message 101
6.5 Failure_Report message 103
6.6 Report_Error message 103

Chapter 7: AIN parameters 105
7.1 Introduction 105
7.2 General requirements 105
7.2.1 AINDigits parameters 105
7.2.2 Trunk group numbers 105
7.3 AccessCode parameter 106
7.4 ACGGlobalOverride parameter 106
7.5 AlternateCarrier parameter 107
7.6 AlternateTrunkGroup parameter 107
7.7 AMADigitsDialedWC parameter 107
7.8 AMAAlternateBillingNumber parameter 108
7.9 AMALineNumber parameter 108
7.10 AMAMeasure parameter 109
7.11 AMAslpID parameter 110

7.12 ApplicationErrorString parameter 110
7.13 BearerCapability parameter 110
7.14 BusyCause parameter 111
7.15 CalledPartyID parameter 111
7.16 CallingPartyID parameter 112
7.17 Carrier parameter 113
7.18 CcID parameter 113
7.19 ChargeNumber parameter 114
7.20 ChargePartyStationType parameter 116
7.21 ClearCause parameter 116
7.22 CloseCause parameter 117
7.23 CollectedAddressInfo parameter 117
7.24 CollectedDigits parameter 117
7.25 CsID parameter 117
7.26 DestinationAddress parameter 118
7.27 DisplayText parameter 118
7.28 EDPNotification parameter 118
7.29 EDPRequest parameter 119
7.30 ExtensionParameter parameter 119
7.31 FailureCause parameter 119
7.32 FeatureActivatorID parameter 119
7.33 ForwardCallIndicator parameter 120
7.34 GenericAddress parameter 120
7.35 GenericAddressList parameter 121
7.36 IPReturnBlock parameter 121
7.37 JurisdictionInformation parameter 122
7.38 Lata parameter 122
7.39 LegID Parameter 122
7.40 OutpulseNumber parameter 123
7.41 PointInCall parameter 124
7.42 PrimaryTrunkGroup parameter 124
7.43 ResourceType parameter 124
7.44 SecondAlternateCarrier parameter 125
7.45 SecondAlternateTrunkGroup parameter 125
7.46 STRConnection parameter 125
7.47 StrParameterBlock parameter 125
7.48 TriggerCriteriaType parameter 126
7.49 UserID parameter 128
7.50 VerticalServiceCode parameter 129

Chapter 8: Extension parameters 131

8.1 Introduction 131
8.2 General requirements 131
8.2.1 Compatibility parameters 131
8.3 Summary of extension parameter identifiers 132
8.4 DMS-250 routing feature compatibility 138
8.4.1 STS and OPART/TPART pairs 138
8.4.2 Call type 139
8.4.3 Satellite restriction 140
8.4.4 Class of service 140
8.5 Authcode compatibility 140

8.5.1 Authcode database index parameter 140
8.6 Billing 141
8.6.1 SCP billing sequence number 141
8.6.2 CDR Field population 141
8.7 Call branding 141
8.7.1 Call branding parameter 141
8.8 Universal Access 141
8.8.1 Universal Access number parameter 141
8.9 PSA SCU support 142
8.9.1 Connect to SCU parameter 142
8.10 Treatment control 142
8.10.1 Treatment code parameter 142
8.11 CAIN trigger subscription 143
8.11.1 CAIN subscription group parameter 143
8.11.2 Call control parameter 143
8.11.3 Subscription Information parameter 143
8.12 Outgoing message enhancements 144
8.12.1 Collected Address Information parameter 144
8.12.2 Originating trunk information parameter 144
8.12.3 Terminating trunk information parameter 144
8.12.4 Busy route parameter 144
8.12.5 Reoriginated call indicator parameter 145
8.12.6 Number of Reoriginations parameter 145
8.12.7 T1 timeout overflow requery indicator parameter 145
8.12.8 Local Number Portability (LNP) information received indicator parameter 146
8.12.9 Jurisdiction Information parameter 146
8.12.10 Switch Identifier parameter 146
8.12.11 Billing Number parameter 146
8.12.12 Account Code parameter 147
8.12.13 PIN Digits parameter 147
8.12.14 Connected Time indicator parameter 148
8.13 Response message enhancements 148
8.13.1 Reorigination allowed indicator parameter 148
8.14 Event Detection Point (EDP) control 148
8.14.1 Network_Busy EDP actions parameter 148
8.14.2 O_Called_Party_Busy EDP actions parameter 149
8.14.3 O_No_Answer EDP actions parameter 149
8.14.4 EDP digit buffer indicator parameter 150
8.14.5 Switch_Hook_Flash Enabled Legs parameter 150
8.15 Multi-switch Business Group support 150
8.15.1 MBG/NETINFO parameter 150
8.16 Global Inter Machine Trunk (IMT) support 151
8.16.1 Universal Translations index parameter 151
8.17 Intelligent Peripheral support 151
8.17.1 Send_To_Resource Connection Type parameter 151
8.18 Virtual IP support 151
8.18.1 Pretranslator Name parameter 151
8.19 CAIN Pretranslator support 152
8.19.1 CAIN Pretranslator error scenario parameter 152
8.20 Code Gapping Support 152

8.20.1 Automatic Code Gapping (ACG) Re-query parameter 152
8.21 Access Type support 152
8.21.1 Access Type parameter 152

Chapter 9: Messaging scenarios 155
9.1 Introduction 155
9.2 FGD call with normal routing 156
9.3 FGD call with default routing 157
9.4 DAL N00 call, disconnected 158
9.5 DAL N00 call, SCP failure 159
9.6 DAL VPN call with conversation 160
9.7 FGD call using the IP in conversation 161
9.8 FGD call with IP interaction cancelled 162
9.9 DAL N00 call with T1 timeout overflow requery 163
9.10 FGD call, Info_Analyzed TDP reevaluation, Office_Code trigger 164
9.11 FGD call with LNP information and direct-termination routing 165
9.12 FGD call with EDP-Request 166
9.13 DAL call with EDP-Notifications 167
9.14 SS7 IMT with an OCM completed transaction followed by a TCM transaction 168
9.15 DAL call with EDP-Notifications and TCM 169
9.16 DAL call with O_Mid_Call and Collect_Information 170
9.17 FGD call with IP Interactions with Call_Info_From_Resource 171
9.18 FGD call with Connect_To_Resource 172
9.19 SS7 FGD EDP call with Office_Code Query 173
9.20 ACG with ACG_Overflow 174
9.21 ACG_Global_Ctrl_Restore with ACG_Global_Ctrl_Restore_Success 175
9.22 Termination_Notification with UA 176
9.23 Termination_Notification with REL message 178
9.24 Termination_Notification with AINF 179
9.25 Multiple Send_Notification with Termination_Notification 181
9.26 Connect_To_Resource from Timeout EDP-Request 183
9.27 Call Transfer via SwitchHookFlash (Scenario 1) 185
9.28 Call Transfer via SwitchHookFlash (Scenario 2) 188

Appendix A: Parameter encodings 191
A.1 Introduction 191
A.2 Encodings for AIN parameters 191
A.2.1 Encoding the Numbering Plan values 191
A.2.2 Encoding the Nortel specific Nature of Address values for Called and Calling Parameters 191
A.2.3 Encoding the Nortel specific Nature of Address values for Charge Number Parameters 192
A.2.4 Encoding the assignment authority 193
A.2.5 Encoding the trigger criteria type 193
A.2.6 Encoding generic address Type of Digits values 194
A.2.7 Encoding maximum digits to collect values 194
A.2.8 Encoding the EDPNotification parameter 195
A.2.9 Encoding the EDPRequest Parameter 195
A.3 Encoding the O_Feature_Requested message 195

A.4 Encodings for extension parameters in the NT_DMS250_EXTENSION_PARAMETER_FAMILY 196
A.4.1 Encoding the STS pair 196
A.4.2 Encoding the primaryTrunkGroupSTS parameter 196
A.4.3 Encoding the alternateTrunkGroupSTS parameter 197
A.4.4 Encoding the secondAlternateTrunkGroupSTS parameter 197
A.4.5 Encoding the overflowRoutingNoSTS parameter 197
A.4.6 Encoding the call type 197
A.4.7 Encoding the satellite restriction 197
A.4.8 Encoding the class of service 197
A.4.9 Encoding the billing sequence number 197
A.4.10 Encoding the call branding announcement 198
A.4.11 Encoding the Universal Access number 198
A.4.12 Encoding the connect to SCU indicator 198
A.4.13 Encoding the CAIN subscription group 199
A.4.14 Encoding the authcode database index 199
A.4.15 Encoding the treatment code 199
A.4.16 Encoding the reoriginated call indicator 199
A.4.17 Encoding the reorigination allowed indicator 199
A.4.18 Encoding the trunk information 199
A.4.19 Encoding the busy route information 200
A.4.20 Encoding the Network_Busy EDP actions 201
A.4.21 Encoding the O_Called_Party_Busy EDP actions 201
A.4.22 Encoding the O_No_Answer EDP actions 202
A.4.23 Encoding the EDP digit buffer indicator 202
A.4.24 Encoding the Universal Translations index 203
A.4.25 Encoding the Multi-switch Business Group (MBG) NETINFO parameter 203
A.4.26 Encoding the T1 timeout overflow re-query indicator 204
A.4.27 Encoding the Local Number Portability (LNP) information received indicator 204
A.4.28 Encoding the Call Control parameter 205
A.4.29 Encoding the amaDigits parameter 205
A.4.30 Encoding the subscriptionInfo parameter 205
A.4.31 Encoding the collectedAddress parameter 206
A.4.32 Encoding the numReorig parameter 206
A.4.33 Encoding the jurisdictionInformation parameter 206
A.4.34 Encoding the switchID parameter 207
A.4.35 Encoding the billingNumber parameter 207
A.4.36 Encoding the accountCode parameter 207
A.4.37 Encoding the pinDigits parameter 207
A.4.38 Encoding the connectTime parameter 207
A.4.39 Encoding the strConnectionType parameter 208
A.4.40 Encoding the pretranslatorName parameter 208
A.4.41 Encoding the cainPRT parameter 208
A.4.42 Encoding the acgRequery parameter 208
A.4.43 Encoding the shfelegs parameter 209
A.5 Encodings for extension parameters in the infoCollectedExtensionAa family 209
A.5.1 Encoding the accessType parameter 209
A.6 Encoding the FlexParameterBlock parameter 209

A.7 Encoding the IPReturnBlock parameter 211

Appendix B: SCCP message protocol 215
B.1 Introduction 215
B.2 SCCP address procedures 215
B.3 Global title use for AIN 216
B.3.1 CLID global title requirements 216
B.3.2 Address global title requirements 217
B.3.3 Feature global title requirements 217
B.3.4 Office code global title requirements 217

Appendix C: ClearCause meaning 219
C.1 Introduction 219
C.2 In-switch Resources 219
C.3 IP Resources 220

List of abbreviations 225

Requirement-Object Index 229
0.1 Cross-Reference for Requirements 229

About this document

Introduction

This document provides an overview of the NetworkBuilder Carrier Advanced Intelligent Network (AIN) Transaction Capabilities Application Part (TCAP) messaging. It defines the message set and protocol used for the interface between the service switching point (SSP) and the service control point (SCP). The NetworkBuilder SSP is implemented within the UCS DMS-250 switch.

This document tailors the AIN TCAP specification in GR-1298-CORE and GR-1299-CORE to meet the needs of the NetworkBuilder product; the tailored version is referred to as NetworkBuilder Carrier AIN (CAIN).

Tailoring of the Bellcore specifications is necessary because the specifications only address the needs of the local exchange carrier (LEC) market. The needs of interexchange carrier (IEC) applications, such as virtual private networks (VPN), are not met. As a result, selected extensions are required as indicated in this document. In addition, portions of the specifications pertaining to LEC-specific features or agents are not applicable and will be excluded from the NetworkBuilder implementation.

Further exclusions may be made for the initial implementations of the NetworkBuilder product which do not fully support the AIN call model and SSP-SCP interface. Adjuncts only are not supported in this release.

Intended audience

This document is intended for use by personnel responsible for planning, developing, maintaining, or testing applications or protocol software that interfaces with NetworkBuilder using Carrier AIN TCAP messaging.

Applicability of this document

This document applies to UCS DMS-250 offices that have UCS14 (CSP14) and have purchased the necessary SOC options. Unless revised, it also applies to offices that have software releases greater than UCS14 (CSP14).

How this document is organized

Chapter 1, “Introduction” provides an overview of AIN concepts.

Chapter 2, “Messaging overview” provides an overview of the messages and parameters supported in this implementation, and includes statements of extensions and exclusions.

Chapters 3–8 provide the rules for populating messages and parameters. Each chapter has a sequence of requirements for the Carrier AIN protocol; these requirements supersede the requirements of GR-1298-CORE and GR-1299-CORE and may either extend or limit those requirements. Deviations from the Bellcore specification of AIN are noted in detail.

Chapter 9, “Messaging scenarios” presents scenarios of the most common SSP-SCP interactions.

Appendix A, “Parameter encodings” provides Abstract Syntax Notation One (ASN.1) descriptions of extension parameters and encoding information for indicators added to existing parameters. Encoding of messages and parameters that match the Bellcore specifications are provided in GR-1299-CORE and are not repeated in this document.

Appendix B, “SCCP message protocol” describes the addressing and routing of TCAP messages via the Service Connection Control Part (SCCP).

Appendix C, “ClearCause meaning” describes the SSP event which causes a given `ClearCause` value to be sent to the SCP.

How to check the version and issue of this document

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

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

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

This document is written for all UCS DMS-250 offices. More than one version of this document may exist. To determine whether you have the latest version of this document, check the release information in *UCS DMS-250 Master Index*, 297-2621-001, or call your UCS DMS-250 Brand Management prime.

References in this document

Bellcore specifications GR-1298-CORE and GR-1299-CORE are incorporated by reference into this document. The implementation of Carrier

AIN TCAP will conform to these specifications except as noted in this document. Access to these Bellcore documents, as well as GR-246-CORE and related documents, is required in order to construct a working SCP TCAP interface to the NetworkBuilder SSP.

The following documents are referred to in this document:

- Bellcore Specification GR-1298-CORE, *AIN 0.2 Switching Systems Generic Requirements,* Issue 4
- Bellcore Specification GR-1299-CORE, *AIN 0.2 Switch-SCP/Adjunct Interface Generic Requirements,* Issue 4
- Bellcore Specification GR-394-CORE, *Switching Systems Generic Requirements for Interexchange Carrier Interconnection Using the Integrated Services Digital Network User Part (ISDNUP),* Issue 1
- CCITT Recommendation X.208, *Specification of Abstract Syntax Notation One (ASN.1),* 1988
- CCITT Recommendation X.209, *Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1),* 1988
- Bellcore Specification GR-246-CORE, *Bell Communications Research Specification for Signalling System Number 7,* Issue 1
- Bellcore Specification GR-2936-CORE, *Local Number Portability Capability Specification*, Issue 1
- Bellcore Specification GR-1129-CORE, *AINGR: Switch Intelligent Peripheral Interface (IPI)*, Issue 2
- *Illinois Number Portability Workshop Generic Switching and Signaling Requirements for Number Portability*, Generic Requirements Issue 1.04
- Bellcore Specification GR-1100-CORE, *Bellcore Automatic Message Accounting Format (BAF) Requirements,* Issue 1, Revision 7
- Bellcore Specification GR-82-CORE, *Signaling Transfer Point (STP) Generic Requirements,* Issue 21

How requirements are presented

A requirement is a feature or function that must be implemented as described in this document in order to meet the needs of the Carrier AIN protocol. Failure to fulfill a requirement may limit the functionality of the application, result in the improper operation of the network, or hinder operations. Requirements are described in imperative language (for example, using the term “shall”).

Within the body of this document, specific requirements are labeled for ease of reference. This practice mirrors that used in recent Bellcore specifications. Each requirement has an absolute label; in addition, one or more requirements may be grouped under a local (chapter-relative) label for cross-reference purposes.

Absolute labels

Absolute labels are applied using a requirement number in brackets (in bold type). For example, the 34th requirement in the document would be numbered "**[34]**". These numbers will not change in subsequent versions of this document; instead, new requirements will be given new numbers regardless of where they appear in the document. Updated requirements will be marked with a revision code, for example, "**[34v2]**" for the first revision of requirement 34.

Local labels

Local labels are applied using the letter "R" and a number composed of the chapter number plus a sequence number local to that chapter. For example, the third requirement found in Chapter 4 would be numbered "**R4-3**". These numbers may change in subsequent versions of this document as new requirements are added.

How AIN identifiers are presented

Monospaced type (`Analyze_Route`) is used for AIN protocol identifiers, such as message names, parameter names, field names, and field values. Capitalization of these identifiers follows the style used in Bellcore specifications such as GR-1299-CORE.

Chapter 1: Introduction

1.1 AIN overview

As described in GR-1298-CORE, the Advanced Intelligent Network (AIN) call model used by NetworkBuilder provides a set of points in call (PIC) to define key states in the processing of an origination or termination at the service switch point (SSP). All NetworkBuilder SSP functions take place within the UCS DMS-250 switch.

Without AIN, a call would proceed linearly through these states using service logic programs (SLP) available on the SSP. With AIN, the AIN service control point (SCP) can take control of the call at one of these PICs, use its own SLPs to provide services for the call in progress, and return control to the SSP at a different PIC, possibly skipping the intervening states.

With AIN, there are two types of events that cause the SSP to communicate with the SCP: triggers and requested events. These events take place at certain detection points during call processing. During an AIN call, the SSP can encounter triggers at trigger detection points (TDPs), and can encounter requested events at event detection points (EDPs).

TDP encounters are determined by in-switch logic and datafill; at each PIC that has an associated TDP, the SSP determines whether the TDP may apply to the call in progress, and whether to trigger the call at that point.

Triggering at a TDP causes the SSP to stop processing the call in progress and send a message to the SCP requesting instructions. This request allows the SCP to take control of the call using its own SLPs, which may include requesting the SSP to perform additional digit collection.

EDPs, on the other hand, are provisioned, or “armed,” by the SCP when the SCP sends a `Request_Report_BCM_Event` component (section 5.3.1) to the SSP in an open transaction initiated by the SSP. This causes the SSP to associate a Next Event List (NEL) with the open transaction. The NEL is a list of EDPs to be armed to send an EDP-Request or an EDP-Notification to the SCP.

When an EDP is encountered as an EDP-Request, the SSP stops processing the call in progress and sends an EDP-Request message to the SCP. As with TDPs, the SCP takes control of the call using its own SLPs and then instructs the SSP as to what action to perform for the call.

When an EDP is encountered as an EDP-Notification, the SSP does not stop call processing, and sends an EDP-Notification message to the SCP. The SCP does not respond to the SSP upon receipt of the EDP-Notification message.

This release only supports ACG transactions originated at the SCP.

TCAP messaging, as defined in this document and applicable Bellcore specifications, is used as the medium for communication between the SSP and SCP.

1.2 Implementation phases

The NetworkBuilder implementation of Carrier AIN on the UCS DMS-250 switch will proceed in several phases. This document is specifically tailored to the UCS14 release, and covers only the features available in that and previous releases. Future releases may change the functionality described here or add to it.

Chapter 2: Messaging overview

2.1 Introduction

The Advanced Intelligent Network (AIN) message and parameter set provides the interface between AIN applications on the service switching point (SSP) and the applications and database on the service control point (SCP).

Bellcore specification GR-1298-CORE describes the AIN call model and the processing requirements for SSP and SCP in response to the TCAP message and parameter set. These requirements are refined for NetworkBuilder. GR-1299-CORE describes the encoding of the messages and parameters, as well as the protocol used to exchange messages and handle error conditions.

2.2 Message types

The message set may be divided into three categories of messages: call-related messages, non-call-related messages, and error messages.

2.2.1 Call-related messages

Call-related messages are used to drive call processing. At certain points in a call, the AIN application on the SSP sends a message to the SCP requesting instructions regarding the disposition of the call in progress (for example, routing information). The SCP may provide the necessary information to the SSP and terminate the dialog, or may extend the dialog by requesting additional information from the SSP before providing final instructions to the SSP. This allows call processing logic to be migrated from the SSP to the SCP, with the SSP acting as a terminal or connection server to the SCP applications.

2.2.2 Non-call related messages

Non-call related messages are used by the SCP to adjust parameters in the SSP. For example, these messages may be used to check trigger settings or disable particular triggers. In addition, a non-call-related message may be used to set up event detection points (EDP) for a call in progress. Both the SSP and SCP may use non-call related messages to control Automatic Code Gapping (ACG). ACG provides the capability to prevent overloading a given SCP and reduce the impact of mass calling events.

2.2.3 Error messages

Error messages are used to respond to error situations, such as receipt of an unexpected communication or the inability of the SSP or SCP to carry out a particular function.

2.3 Exclusions and limitations

Only a selected portion of the message and parameter set described in GR-1299-CORE is implemented for NetworkBuilder. The following sections describe the specific messages and parameters from GR-1299-CORE available in this implementation.

2.3.1 Message transport

R2-1 **[1v2]** TCAP messaging is supported only on ANSI SCCP/MTP.

R2-2 **[2]** TCAP messages to and from the SSP shall be limited to a single unitdata (UDT) SCCP message. Segmented messages (extended unitdata [XUDT] messages) are not supported.

R2-3 **[3]** The UCS14 release is based on ANSI Issue 2 TCAP (ANSI T1.114-1992). TCAP dialog portion shall not be required.

R2-4 **[4v3]** Only one call-related component shall be sent in any TCAP package sent from the SSP. The SSP may receive a TCAP package containing a call-related component along with up to four non-call related components. Please refer to Chapter 5, “Non-call Related Messages” for more information.

2.3.2 Message set restrictions

As the NetworkBuilder implementation of Carrier AIN provides only a limited part of the AIN call model described in GR-1298-CORE, a number of messages have been excluded.

For a summary of the messages, see Tables 5-1 and 5-2 in Bellcore specification GR-1299-CORE. The TCAP packaging information for these parameters is summarized in Table 3-7 of that document.

R2-5 **[5v5]** Supported call-related messages from the originating call model, sent by the SSP are shown in Table 2-1.

Table 2-1 Summary of SSP originating call-related messages

Message name	Type			Paragraph reference		
	TDP-R	EDP-R	EDP-N	This doc	GR-1298-CORE	GR-1299-CORE
Call_Info_From_Resource				3.17	N/A	5.2.1.1
Close				3.3	4.5.5	5.2.1.2
CTR_Clear				3.19	21.6.2.6	5.2.1.3
Failure_Outcome				3.22	21.4.13.5	5.2.1.4
Info_Analyzed	x			3.4	4.5.3.3	5.2.1.5
Info_Collected	x			3.5	4.5.3.2	5.2.1.6
Network_Busy	x	x		3.6	4.5.3.4	5.2.1.7
O_Abandon		x		3.21	21.4.9.1	5.2.1.8
O_Answer			x	3.8	4.5.3.7	5.2.1.9
O_Called_Party_Busy	x	x		3.9	4.5.3.6	5.2.1.10
O_Disconnect		x	x	3.15	4.5.3.10	5.2.1.11
O_Feature_Requested	x			3.10	N/A	N/A
O_Mid_Call	x	x		3.13	21.4.6	5.2.1.12
O_No_Answer	x	x		3.11	4.5.3.8	5.2.1.13
O_Term_Seized			x	3.12	4.5.3.5	5.2.1.15
Origination_Attempt	x			3.7	4.5.3.1	5.2.1.16
Resource_Clear				3.18	5.5	5.2.1.17
Timeout		x		3.16	21.4.8	5.2.1.26

R2-6 **[6v2]** Supported call-related messages from the terminating call model, sent by the SSP are shown in Table 2-2.

Table 2-2 Summary of SSP terminating call-related messages

Message name	Type			Paragraph reference		
	TDP-R	EDP-R	EDP-N	This doc	GR-1298-CORE	GR-1299-CORE
Termination_Attempt	x			3.14	4.5.4.1	5.2.1.24

R2-7 **[7v4]** Supported call-related messages sent by the SCP are shown in Table 2-3.

Table 2-3 Summary of SCP call-related messages

Message name	Paragraph reference		
	This doc	GR-1298-CORE	GR-1299-CORE
Acknowledge	4.15	21.6.8	5.2.2.1
Analyze_Route	4.3	5.2.1	5.2.2.2
Authorize_Termination	4.8	5.2.4	5.2.2.3
Call_Info_To_Resource	4.11	N/A	5.2.2.4
Cancel_Resource_Event	4.4	5.5	5.2.2.5
Collect_Information	4.10	5.2.3	5.2.2.6
Connect_To_Resource	4.9	21.6.2	5.2.2.7
Continue	4.5	5.2.6	5.2.2.8
Disconnect	4.6	5.2.7	5.2.2.10
Disconnect_Leg	4.14	21.6.7	5.2.2.11
Merge_Call	4.13	21.6.6	5.2.2.13
Originate_Call	4.12	21.6.1	5.2.2.16
Send_To_Resource	4.7	5.5	5.2.2.18

R2-8 **[645]** Supported non-call related messages sent by the SSP are shown in Table 2-4.

Table 2-4 Summary of Switch non-call related messages

Message name	Paragraph reference		
	This doc	GR-1298-CORE	GR-1299-CORE
ACG_Global_Ctrl_Restore_Success	5.4.3	14	5.3.3
ACG_Overflow	5.4.2	14	5.3.4
Termination_Notification	5.4.1	5.8	5.3.19

R2-9 **[8v4]** Supported non-call related messages sent by the SCP are shown in Table 2-5.

Table 2-5 Summary of SCP non-call related messages

Message name	Paragraph reference		
	This doc	GR-1298-CORE	GR-1299-CORE
ACG	5.3.3	14	5.3.1
ACG_Global_Ctrl_Restore	5.3.4	14	5.3.2
Furnish_AMA_Information	5.3.5	9.2.1	5.3.8
Request_Report_BCM_Event	5.3.1	5.3	5.3.16
Send_Notification	5.3.2	5.8	5.3.17

R2-10 **[9]** The supported AIN error messages are shown in Table 2-6. TCAP error messages (Abort and Reject) are described in GR-246-CORE.

Table 2-6 Summary of error messages

Message name	Sent by		Paragraph reference		
	SSP	SCP	This doc	GR-1298-CORE	GR-1299-CORE
Application_Error	x	x	6.3	7	5.4.1.1
Close	x	x	6.4	4.5.5	5.2.1.2
Failure_Report	x	x	6.5	7	5.4.1.3
Report_Error	x	x	6.6	7	5.4.1.2

Table 2-7 summarizes the introduction history of messages to the CAIN protocol.

Table 2-7 Message History (page 1 of 3)

Message name	Initially supported in UCS
ACG	UCS09
ACG_Global_Ctrl_Restore	UCS09
ACG_Global_Ctrl_Restore_Success	UCS09
ACG_Overflow	UCS09

Message name	Initially supported in UCS
Acknowledge	UCS11
Analyze_Route	UCS05
Application_Error	UCS05
Authorize_Termination	UCS08
Call_Info_From_Resource	UCS08
Call_Info_To_Resource	UCS08
Cancel_Resource_Event	UCS06
Close	UCS05
Collect_Information	UCS08
Connect_To_Resource	UCS08
Continue	UCS05
CTR_Clear	UCS08
Disconnect	UCS05
Disconnect_Leg	UCS11
Failure_Outcome	UCS11
Failure_Report	UCS05
Furnish_AMA_Information	UCS14
Info_Analyzed	UCS05
Info_Collected	UCS06
Merge_Call	UCS11
Network_Busy	UCS06
O_Abandon	UCS11
O_Answer	UCS07
O_Called_Party_Busy	UCS06
O_Disconnect	UCS08
O_Feature_Requested	UCS06
O_Mid_Call	UCS08
O_No_Answer	UCS06
O_Term_Seized	UCS07
Originate_Call	UCS11
Origination_Attempt	UCS06

Message name	Initially supported in UCS
Resource_Clear	UCS06
Report_Error	UCS05
Request_Report_BCM_Event	UCS07
Send_To_Resource	UCS05
Send_Notification	UCS09
Termination_Notification	UCS09
Timeout	UCS08

2.3.3 Parameter set restrictions

A limited set of AIN parameters (as specified in GR-1299-CORE) are provided (see Table 2-11 through Table 2-17 for mapping to messages). A parameter not listed in the above mentioned tables is not supported by NetworkBuilder. If no reference within Chapter 7 is made to a listed parameter, then the use of that parameter in Carrier AIN is unchanged from the Bellcore specifications.

R2-11 [10v6] The supported parameters are shown in Table 2-8.

Table 2-8 Summary of parameters (page 1 of 3)

Parameter name	Paragraph reference		Initially introduced
	This doc	GR-1299-CORE	
AccessCode	7.3	6.1	UCS05
ACGEncountered	N/A	6.2	UCS09
ACGGlobalOverride	7.4	6.3	UCS09
AlternateCarrier	7.5	6.7	UCS07
AlternateTrunkGroup	7.6	6.9	UCS05
AMAAlternateBillingNumber	7.8	6.10	UCS11
AMABAFModules	N/A	6.11	UCS14
AMADigitsDialedWC	7.7	6.14	UCS06
AMALineNumber	7.9	6.15	UCS11
AMAMeasure	7.10	6.16	UCS08
AMASetHexABIndicator	N/A	6.19	UCS14
AMAslpID	7.11	6.21	UCS09

Parameter name	Paragraph reference		Initially introduced
	This doc	GR-1299-CORE	
AnswerIndicator	N/A	6.24	UCS05
ApplicationErrorString	7.12	6.25	UCS05
BearerCapability	7.13	6.28	UCS05
BusyCause	7.14	6.31	UCS06
CalledPartyID	7.15	6.33	UCS05
CallingPartyID	7.16	6.36	UCS05
Carrier	7.17	6.37	UCS05
CcID	7.18	6.40	UCS08
ChargeNumber	7.19	6.41	UCS05
ChargePartyStationType	7.20	6.42	UCS07
ClearCause	7.21	6.43	UCS06
ClearCauseData	N/A	6.44	UCS08
CloseCause	7.22	6.45	UCS05
CollectedAddressInfo	7.23	6.46	UCS05
CollectedDigits	7.24	6.47	UCS05
ConnectTime	N/A	6.49	UCS09
ControlCauseIndicator	N/A	6.50	UCS09
CsID	7.25	6.53	UCS11
DestinationAddress	7.26	6.55	UCS06
DisconnectFlag	N/A	6.57	UCS05
DisplayText	7.27	6.58	UCS08
EchoData	N/A	6.60	UCS09
EDPNotification	7.28	6.62	UCS07
EDPRequest	7.29	6.63	UCS07
ExtensionParameter	7.30	6.68	UCS05
FailureCause	7.31	6.73	UCS05
FeatureActivatorID	7.32	6.75	UCS06
ForwardCallIndicator	7.33	6.76	UCS07
GapDuration	7.34	6.77	UCS09

Parameter name	Paragraph reference		Initially introduced
	This doc	GR-1299-CORE	
GapInterval	7.34	6.78	UCS09
GenericAddress	7.34	6.79	UCS05
GenericAddressList	7.35	6.80	UCS05
GlobalTitleAddress	N/A	6.84	UCS09
IPReturnBlock	7.36	6.86	UCS08
JurisdictionInformation	7.37	6.87	UCS08
Lata	7.38	6.89	UCS11
LegID	7.39	6.90	UCS08
NotificationIndicator	N/A	6.94	UCS07
ONoAnswerTimer	N/A	6.95	UCS07
OutpulseNumber	7.40	6.98	UCS05
OverflowBillingIndicator	N/A	6.99	UCS14
PointInCall	7.41	6.101	UCS06
PrimaryBillingIndicator	N/A	6.103	UCS14
PrimaryTrunkGroup	7.42	6.104	UCS05
ResourceType	7.43	6.108	UCS05
SecondAlternateCarrier	N/A	6.114	UCS07
SecondAlternateTrunkGroup	7.45	6.115	UCS05
StrParameterBlock	7.47	6.124	UCS05
TerminationIndicator	N/A	6.128	UCS09
TimeoutTimer	N/A	6.129	UCS07
TNoAnswerTimer	N/A	6.130	UCS07
TranslationType	N/A	6.131	UCS09
TriggerCriteriaType	7.48	6.133	UCS05
UserID	7.49	6.134	UCS05
VerticalServiceCode	7.50	6.135	UCS06

2.3.4 Message responses

R2-12 **[11v4]** The call-control messages that may be returned from the SCP for each SSP call-related request message are shown in Table 2-9.

Table 2-9 Summary of response messages for SSP call-related requests

Message (x = allowed)	**Analyze_Route**	**Continue**	**Disconnect**	**Send_To_Resource**	**Connect_To_Resource**	**Authorize_Termination**	**Collect_Information**	**Call_Info_To_Resource**	**Originate_Call**	**Merge_Call**	**Disconnect_Leg**	**Acknowledge**
Failure_Outcome			x							x		
Info_Analyzed	x	x	x	x								
Info_Collected	x		x	x								
Network_Busy	x		x	x								
Origination_Attempt	x	x	x	x								
O_Abandon			x							x	x	
O_Called_Party_Busy	x	x		x								
O_Disconnect			x		x					x	x	
O_Feature_Requested	x		x	x								
O_Mid_Call (EDP-R)			x		x				x	x	x	x
O_Mid_Call (TDP-R)	x		x		x		x					
O_No_Answer	x	x		x								
Termination_Attempt			x	x		x						
Timeout			x		x							

R2-13 **[764v3]** The call-control messages that may be returned from the SCP for each SSP call-related response message are shown in Table 2-10.

Table 2-10 Summary of response messages for SSP call-related responses

Message (x = allowed)	**Analyze_Route**	**Continue**	**Disconnect**	**Send_To_Resource**	**Connect_To_Resource**	**Authorize_Termination**	**Collect_Information**	**Call_Info_To_Resource**	**Originate_Call**	**Merge_Call**	**Disconnect_Leg**	**Acknowledge**
CTR_Clear	x	x	x		x		x		x	x	x	x
Resource_Clear	x	x	x	x								
Close												
Call_Info_From_Resource								x				

2.4 Parameter mapping

The following tables summarize the parameters that are present in the TCAP messages sent to and from the SSP. A parameter that is supported for a particular message is marked as either optional or mandatory.

The SSP shall always send any parameters that are marked as mandatory by this protocol definition. The absence of a mandatory parameter in any SSP outgoing message should be considered a fatal application error.

R2-14 **[12v6]** Table 2-11 through Table 2-17 summarize the parameters that may be present in each message.

Table 2-11 Summary of parameters for SSP call-related messages (Part 1) (page 1 of 2)

Parameter Name (m = mandatory, o = optional)	**Info_Analyzed**	**Info_Collected**	**Network_Busy**	**Origination_Attempt**	**O_Answer**	**O_Called_Party_Busy**	**O_Feature_Requested**	**O_No_Answer**	**O_Term_Seized**	**Resource_Clear**	**Close**
AccessCode	o	o					o				
ACGEncountered	o	o	o	o		o	o	o			
BearerCapability	m	m	m	m	m	m	m	m	m		o
BusyCause			o			o					
CalledPartyID	o		o			o		o			
CallingPartyID	o	o	o	o		o	o	o			
Carrier	o	o	o	o		o	o	o			
ChargeNumber	o	o	o	o		o	o	o			
ChargePartyStationType	o	o	o			o	o	o			
ClearCause										m	
ClearCauseData										o	
CloseCause											m
CollectedAddressInfo	o	o					o				
CollectedDigits	o	o					o			o	
ExtensionParameter	o	o	o	o	o	o	o	o	o	o	o
FailureCause										o	
FeatureActivatorID							m				
Lata	o	o	o	o		o	o	o			
NotificationIndicator			o		o	o		o	o		
IPReturnBlock										o	
JurisdictionInformation	o										
PointInCall							m				
TriggerCriteriaType	o	o	o	o		o	o	o			

—continued—Summary of parameters for SSP call-related messages (Part 1)(page 2 of 2)

Parameter Name (m = mandatory, o = optional)	**Info_Analyzed**	**Info_Collected**	**Network_Busy**	**Origination_Attempt**	**O_Answer**	**O_Called_Party_Busy**	**O_Feature_Requested**	**O_No_Answer**	**O_Term_Seized**	**Resource_Clear**	**Close**
UserID	m	m	m	m	m	m	m	m	m		o
VerticalServiceCode	o	o					o				

Table 2-12 Summary of parameters for SSP call-related messages (Part 2)

Parameter Name (m = mandatory, o = optional)	**O_Mid_Call**	**Termination_Attempt**	**Timeout**	**O_Disconnect**	**Call_Info_From_Resource**	**CTR_Clear**	**O_Abandon**	**Failure_Outcome**
AccessCode	o							
ACGEncountered	o	o						
BearerCapability	m	m	m	m			m	m
CalledPartyID	o	o						
CallingPartyID	o	o						
Carrier	o							
CcID	o		o			o	o	o
ChargeNumber	o	o						
ChargePartyStationType	o	o						
ClearCause						m		
CollectedDigits						o		
ExtensionParameter	o	o	o	o				
FailureCause						o		o
FeatureActivatorID	o							
NotificationIndicator	o		o	o			o	
IPReturnBlock					o	o		
Lata		o						
LegID				o		o		o
PointInCall	o			o			o	o
TriggerCriteriaType	o	o						
UserID	m	m	m	m			m	m
VerticalServiceCode	o							

Table 2-13 Summary of parameters for SCP call-related response messages (Page 1 of 2)

Parameter Name (m = mandatory, o = optional)	**Analyze_Route**	**Cancel_Resource_Event**	**Continue**	**Disconnect**	**Send_To_Resource**	**Authorize_Termination**	**Connect_To_Resource**	**Call_Info_To_Resource**	**Collect_Information**	**Originate_Call**	**Merge_Call**	**Disconnect_Leg**	**Acknowledge**
AlternateCarrier	o									o			
AlternateTrunkGroup	o									o			
AMAAlternateBillingNumber	o		o	o	o	o	o		o	o			
AMADigitsDialedWC	o		o	o	o	o	o		o	o			
AMALineNumber	o		o	o	o	o	o		o	o			
AMAMeasure					o		o						
AMAslpID	o				o					o			
AnswerIndicator					o								
CalledPartyID	o									o			
CallingPartyID	o					o			o	o			
Carrier	o									o			
ChargeNumber	o									o			
ChargePartyStationType	o									o			
CsID												o	o
DestinationAddress					o		o						
DisconnectFlag					o		o						
DisplayText						o				o			
ExtensionParameter	o	o	o	o	o	o	o	o	o	o		o	o
ForwardCallIndicator	o									o			
GenericAddressList	o									o			
LegID							o					m	
OutpulseNumber	o									o			
OverflowBillingIndicator	o												

Parameter Name (m = mandatory, o = optional)	**Analyze_Route**	**Cancel_Resource_Event**	**Continue**	**Disconnect**	**Send_To_Resource**	**Authorize_Termination**	**Connect_To_Resource**	**Call_Info_To_Resource**	**Collect_Information**	**Originate_Call**	**Merge_Call**	**Disconnect_Leg**	**Acknowledge**
PrimaryBillingIndicator					o								
PrimaryTrunkGroup	o									o			
ResourceType					m		m	o					
SecondAlternateCarrier	o									o			
SecondAlternateTrunkGroup	o									o			
StrParameterBlock					m		m	o					

Table 2-14 Summary of parameters for SSP non-call-related messages

Parameter name (m = mandatory, o = optional)	**ACG_Overflow**	**ACG_Global_Ctrl_Restore_Success**	**Termination_Notification**
BusyCause			o
ConnectTime			o
ControlCauseIndicator	m		
EchoData			m
ExtensionParameter	o	o	o
GlobalTitleAddress	m		
TerminationIndicator			m
TranslationType	m		

Table 2-15 Summary of parameters for SCP non-call-related messages

Parameter name (m = mandatory, o = optional)	**ACG**	**ACG_Global_Ctrl_Restore**	**Furnish_AMA_Information**	**Request_Report_BCM_Event**	**Send_Notification**
ACGGlobalOverride		m			
AMABAFModules			o		
AMASetHexABIndicator			o		
ControlCauseIndicator	m				
EchoData					m
EDPNotification				o	
EDPRequest				o	
ExtensionParameter	o	o	o	o	o
GapDuration	m				
GapInterval	m				
GlobalTitleAddress	m				
ONoAnswerTimer				o	
TimeoutTimer				o	
TranslationType	m				

Table 2-16 Summary of parameters for error messages

Parameter name (m = mandatory, o = optional)	**Close**	**Report_Error**	**Application_Error**	**Failure_Report**
ApplicationErrorString		m	m	
BearerCapability	o			
CloseCause	o			
ExtensionParameter	o	o	o	o
FailureCause				m
UserID	o			

Table 2-17 Summary of parameters that are ignored upon receipt (page 1 of 2)

Parameter name (i = ignored)	**Analyze_Route**	**Authorize_Termination**	**Call_Info_To_Resource**	**Cancel_Resource_Event**	**Collect_Information**	**Continue**	**Disconnect**	**Furnish_AMA_Information**	**Send_To_Resource**	**Connect_To_Resource**	**Request_Report_BCM_Event**	**Originate_Call**	**Merge_Call**	**Disconnect_Leg**	**Acknowledge**	**ACG**	**ACG_Global_Ctrl_Resotre**	**Send_Notification**
AlternateBillingIndicator	i		i	i	i							i						
AlternateDialingPlanInd			i	i	i							i						
AMABIllingFeature	i	i			i	i	i		i	i		i						
AMABusinessCustomerID	i	i				i	i		i	i		i	i					
AMADigitsDialedWC													i					
AMASequenceNumber	i	i			i	i	i		i	i		i						
AMAServiceProviderID	i	i			i	i	i		i	i		i						
AMAslpID													i					
Amp1	i	i	i	i	i	i	i		i	i		i	i		i			
Amp2	i	i	i	i	i	i	i		i	i		i	i		i			
BearerCapability												i	i					
CallingPartyBGID	i											i						
CarrierUsage	i											i						
CollectedDigits					i							i						
ControllingLegTreatment		i										i						
DisplayText													i	i				
DPConverter					i				i	i		i						
ExtendedRinging					i					i								
GenericDigitsList	i											i						
LampTreatment												i		i				
NetworkSpecificFacilities	i											i						
OverflowBillingIndicator					i							i						
PassiveLegTreatment	i											i		i				

Parameter name (i = ignored)	**Analyze_Route**	**Authorize_Termination**	**Call_Info_To_Resource**	**Cancel_Resource_Event**	**Collect_Information**	**Continue**	**Disconnect**	**Furnish_AMA_Information**	**Send_To_Resource**	**Connect_To_Resource**	**Request_Report_BCM_Event**	**Originate_Call**	**Merge_Call**	**Disconnect_Leg**	**Acknowledge**	**ACG**	**ACG_Global_Ctrl_Resotre**	**Send_Notification**
Prefix	i																	
PrimaryBillingIndicator	i	i			i	i	i			i		i						
RedirectingPartyID	i											i						
RedirectionInformation	i																	
RTPReroutingNumber							i											
SecondAlternateBillingIndicator	i				i							i						
ServiceContext	i	i	i	i	i	i	i		i	i		i						
ServiceProviderID	i	i	i	i	i	i	i		i	i		i	i					
TNoAnswerTimer											i							
Tcm	i	i										i						

2.5 Extensions to AIN TCAP

The following paragraphs provide an overview of NetworkBuilder extensions to the AIN TCAP specification in GR-1299-CORE which constitute Carrier AIN.

2.5.1 Originating Feature Requested trigger

Carrier AIN brings forward from AIN 1.0 the `O_Feature_Requested` TDP and trigger for the `Collect_Information` PIC. For NetworkBuilder, the TDP is encountered immediately after address collection. The parameters supported for this SSP request message are similar to those supported for the `Info_Collected` request message.

2.5.2 AINDigits parameters

The AIN specification provides a limited range of identifiers for the Nature of Address (NOA) field of parameters containing address or billing number strings. In the NetworkBuilder implementation, extensions are required to describe additional types of numbers, such as authorization codes, which do not apply in the local exchange carrier (LEC) environment encompassed by Bellcore specifications. This implementation adds NOA values to existing

digits parameters, and adds new Type of Digits identifiers to existing generic address parameters.

In addition, the number of digits to be transmitted in these parameters is not limited by the ranges given for the parameter in GR-1299-CORE, as the Bellcore specifications for populating digits parameters in GR-1298-CORE are predicated on LEC numbering schemes that may not always apply in the NetworkBuilder environment.

2.5.3 Other AIN parameters

The following additional AIN parameters are extended:

- The sequence of `AMADigitsDialedWC` parameters used in certain SCP response messages is increased in size to a maximum of 7 parameters.
- The `AnnounceElement` subparameter used in the `Send_To_Resource` and `Connect_To_Resource` message is extended to allow the SCP to direct a variable number of digits to be collected with an explicit maximum, in addition to the existing capability to collect a variable number of digits until a timeout occurs.
- Reserved values are allowed in the `BearerCapability` parameter (7kHz audio and multi-rate), except in queries resulting from the Office_Code trigger. These reserved values are only sent *if the CAIN protocol version is* V2 *or lower.*
- The `TriggerCriteriaType` parameter may also use reserved and spare values based on provisioning in the trigger tables.
- The `UserID` parameter may contain the switch ID and trunk group number of the caller. Please refer to sections 2.7, 3.2.2 and 7.49 for more information.
- The `VerticalServiceCode` parameter is extended by allowing the feature digit to be “#” as well as “*”.
- The `BusyCause` parameter is supported within the `Network_Busy` message. Please refer to Chapter 3.6, “Network_Busy message” for more information.

2.5.4 Extension parameters

Carrier AIN supports an extended parameter set through the use of the AIN `ExtensionParameter` parameter. This extended parameter set is provided to convey optional feature-specific information that may not easily be transmitted within the standard AIN parameter set.

The NetworkBuilder implementation does not assume the existence of an SCP capable of processing or returning parameters from the extended set; all basic AIN call-control functions are capable of operating in the absence of extension parameters. However, parts of the UCS DMS-250 feature set may have limited

application without the extended parameter set detailed in Chapter 7, “AIN parameters”.

Only the extension parameters described within this document are supported. These parameters are identified using the Nortel Networks Carrier AIN assignment authority identifier. Other extended parameter sets may be supported only upon prior arrangement with Nortel Networks.

R2-15 **[13]** The Carrier AIN assignment authority object identifier is

```
cain02ProtocolDefn OBJECT IDENTIFIER ::=
    { iso member-body usa(840) nt(113533)
          iec(8) cain02(65) protocolDefn(16) }
```

See section A.2.4 for the encoding of the assignment authority object identifier.

2.6 Office Code trigger

The Office Code trigger in the Info_Analyzed TDP allows the querying of a Local Number Portability (LNP) SCP to determine if the called address is a ported number. Because the LNP SCP is based on the AIN 0.1 specification, queries resulting from the Office_Code trigger must be AIN 0.1 compliant. Therefore, NetworkBuilder AIN 0.2 extensions to the `Info_Analyzed` message do not apply for queries resulting from the Office_Code trigger. For more information regarding LNP, refer to Illinois Number Portability Workshop Generic Switching and Signaling Requirements for Number Portability, Generic Requirements Issue 1.04.

2.7 Protocol version control

Over time, messages and parameters from earlier versions of this protocol definition may require changes which are not fully upwardly-compatible.

In UCS06 and beyond, the CAIN protocol version is controlled by in-switch datafill. The value of the CAIN protocol version determines the contents and the format of outgoing messages sent from the SSP. The CAIN protocol version consists of two parts: the stream (e.g. - UCS05, UCS06, etc.) and the version (e.g. - v0, v1, etc.). It is expected that new streams and versions will include the features of the previous streams and versions.

The stream designation of the protocol version determines whether certain messages or parameters are allowed to be sent in outgoing SSP TCAP packages:

Table 2-18 Protocol version control based on stream designation (page 1 of 2)

Stream	controlled functionality
UCS05	The parameters allowed in outgoing SSP messages by later protocol version streams are not supported for this stream designation.
UCS06	The `VerticalServiceCode` parameter is allowed in the `Info_Analyzed` message (see R3-47 in section 3.4). The `adin`, `treatment`, `cainGroup`, `origTrunkInfo`, and `reorigCall` extension parameters are allowed in the `Info_Analyzed` message (see sections 8.5.1, 8.10.1, 8.11.1, 8.12.1, and 8.12.5).
UCS07	The `lnpReceived` extension parameter is allowed in the `Info_Analyzed`, `Info_Collected`, and `O_Feature_Requested` messages (see R8-67 in Chapter 8, “Extension parameters”) If the `ChargeNumber` parameter has been populated with an ANI, then the Nature of Address shall be national, and the `ChargePartyStationType` parameter shall contain the Information Digits from the call (see R3-38 and R3-42 in section 3.4).
UCS08	The `subscriptionInfo` extension parameter is allowed in all TDP-Request messages. The `cainPRT` extension parameter is allowed in the `O_Feature_Requested` message. The `JurisdictionInformation` parameter is allowed in the `Info_Analyzed` message. The `jurisdictionInformation` extension parameter is allowed in all TDP-Request messages except `Info_Analyzed`. The `collectedAddress` extension parameter is allowed in the `Network_Busy`, `O_Called_Party_Busy`, and `O_No_Answer` TDP-Request messages.

Stream	controlled functionality
UCS09	The `switchID` extension parameter is allowed in all TDP-Request messages. The `accountCode` extension parameter is allowed in `Info_Analyzed`, `Info_Collected`, `O_Feature_Requested`, and `O_Mid_Call` TDP-Request messages. The `billingNumber` extension parameter is allowed in all TDP-Request messages. The `pinDigits` extension parameter is allowed in `Info_Analyzed`, `Info_Collected`, and `O_Feature_Requested` TDP-Request messages.
UCS11	CAIN02 `Lata` parameter is added as an optional parameter for requests outgoing from the SSP.

The version designation of the protocol version indicates the encoding format to use for parameters whose encodings have changed between software releases:

Table 2-19 Protocol version control based on version designation (page 1 of 2)

Version	controlled functionality
V0	The `universalAccess` extension parameter allows up to 10 digits to be sent to the SCP (see sections 8.8.1 and A.4.11).
V1	The `universalAccess` extension parameter allows up to 24 digits to be sent to the SCP (see sections 8.8.1 and A.4.11).
V2	The population of the Nature of Address and Numbering Plan of the `CollectedAddressInfo` parameter shall conform to GR-1298-CORE requirements [189], [190], and [191]). In previous versions, the `CollectedAddressInfo` Nature of Address and Numbering Plan were set to unknown (see this document R7-51 ([106v3]) in section 7.23).

Version	controlled functionality
V3	The UserID parameter can take the TrunkGroup or PrivateFacilityGID formats as specified in GR-1299-CORE. See “7.49 UserID parameter” on page 128. The AccessCode standard parameter is no longer populated. The CollectedDigits parameter is not used to transmit PIN digits to the SCP. The interpretation of the AMADigitsDialedWC is aligned with GR-1100-CORE. Non-standard NOAs from the outgoing CalledPartyID parameter are removed. Additional standard NOAs for the outgoing CalledPartyID parameter are added. Non-standard bearer capability values are removed from the BearerCapability parameter.
V4	Additional standard NOAs for the outgoing ChargeNumber parameter are added. Additional standard NOAs for the outgoing CollectedAddressInfo parameter are added.
V5	Population of the CallingPartyID parameter outgoing from the SSP is modified. See “3.2.4 Calling party identification” on page 31.

The CAIN protocol version, as determined by in-switch datafill, is a combination of stream designation and version. For example, if the CAIN protocol stream is designated UCS06, the sending of the VerticalServiceCode parameter and the adin, treatment, and reorigCall extension parameters in the Info_Analyzed message (if appropriate to the call scenario) is allowed. When the CAIN protocol version is V1 or higher and the universalAccess extension parameter is included, it may contain up to 24 digits.

R2-16 [318] (deleted)

Chapter 3: SSP call-related messages

3.1 Introduction

This chapter provides an overview of the service switching point (SSP) call-related messages supported by the NetworkBuilder implementation of Carrier AIN, together with rules for populating the messages with parameters. Call-related messages are those that direct call control for a call in progress. This chapter overrides portions of section 4.5 of GR-1298-CORE. Encodings for all messages and parameters are provided in Appendix A or in GR-1299-CORE.

Actions required on the part of the SSP or service control point (SCP) in generating or responding to these messages are described in GR-1298-CORE. Refer to GR-1298-CORE for full requirements on message processing, except where explicitly overridden in the following descriptions.

For each message described in this chapter, a table shows the expected parameters (in order) together with the nature of data to be provided in that parameter. Mandatory parameters are marked in **bold**. Parameters not listed are not supported. Refer to Table 2-17 for more information.

3.2 General requirements

3.2.1 Messages

The requirements described in the following paragraphs apply to all messages.

R3-1 **[14v4]** The current NetworkBuilder call model provides support for two-party calls and limited support for three-party calls.

R3-2 **[15]** The population rules for messages in this chapter shall override those in GR-1298-CORE and GR-1299-CORE. Mandatory parameters are marked in **bold**.

R3-3 **[16]** Extension parameters shall always be optional. Any information transferred in the extension parameter shall be optional; NetworkBuilder shall not require the use of an extension parameter to provide basic trunk-to-trunk call functionality. This allows core functionality to operate in the situation where the SSP communicates with an SCP that cannot support the extension parameters described in Chapter 8, “Extension parameters”.

3.2.2 UserID parameter

The `UserID` parameter is used to inform the SCP of the network identity of the originator for OCM requests and the terminator for the TCM. NetworkBuilder supports three encodings of the UserID as specified in GR-1299-CORE. When the UserID parameter is encoded in the DN format, the switch ID shall be in the range 0–127. The 5-digit trunk number is based on the trunk group's value in inswitch datafill and is unique to each trunk group in the switch. Values from 0–99999 are supported, which exceeds the current UCS DMS-250 switch internal limits.

R3-4 **[17v3]** The `UserID` parameter shall take the `DN` form in all messages sent by the SSP, except for messages resulting from the `Office_Code` trigger *only if the version designation of the CAIN protocol version is less than* V3.

R3-5 **[319v2]** The `UserID` parameter shall take the `TrunkGroupID` form in messages resulting from the `Office_Code` trigger *only if parameter PRIVATE_FACILITY_GROUP_USERID is set to N in table CAINPARM.*

R3-6 **[854]** The `UserID` parameter shall take the `PrivateFacilityGID` form in messages resulting from the `Office_Code` trigger *only if parameter PRIVATE_FACILITY_GROUP_USERID is set to Y in table CAINPARM.*

R3-7 **[449]** The directory number is actually the switch ID and trunk group encoded as described in section 7.49 *only if the version designation of the CAIN protocol version is less than* V3.

R3-8 **[450v2]** The `UserID` parameter shall take the `TrunkGroupID` form in all messages sent by the SSP *only if the CAIN protocol version is* V3 *or greater and parameter PRIVATE_FACILITY_GROUP_USERID is set to N in table CAINPARM.*

R3-9 **[855]** The `UserID` parameter shall take the `PrivateFacilityGID` form in all messages sent by the SSP *only if the CAIN protocol version is* V3 *or greater and parameter PRIVATE_FACILITY_GROUP_USERID is set to Y in table CAINPARM.*

R3-10 **[451v2]** The `UserID` parameter shall indicate the originator for OCM requests and the terminator for TCM requests when in a two-party call. It will indicate the controlling leg (leg 0) when in a three-party call.

3.2.3 Billing numbers

In order to transmit the wider variety of billing and authorization numbers that may be used to validate a subscriber's access to the long distance network, the following scheme shall be used:

R3-11 **[18v4]** (Deleted)

Nortel Networks Proprietary and Confidential

R3-12 **[646]** (Deleted)

R3-13 **[647]** When the SSP has identified a principal billing number — credit card number, N00 number, or authcode — that number shall be transmitted to the SCP in the `billingNumber` extension parameter *only if the CAIN protocol stream is UCS09 or higher.* (Not sent in `Office_Code` queries)

R3-14 **[19v2]** If an account code is required in addition to the principal billing number, that code shall be transmitted in the `AccessCode` parameter *only if the CAIN protocol version is* V2 *or lower.*

R3-15 **[648]** If an account code is required in addition to the principal billing number, that code shall be transmitted in the `accountCode` extension parameter *only if the CAIN protocol stream is* UCS09 *or higher.*

R3-16 **[20v3]** If a personal identification number (PIN) is required, that number shall be transmitted in the `CollectedDigits` parameter, with the Nature of Address field indicating the transmission of PIN digits *only if the CAIN protocol version is* V2 *or lower.*

R3-17 **[649]** If a personal identification number (PIN) is required in addition to either or both of the above, that number shall be transmitted in the `pinDigits` extension parameter *only if the CAIN protocol stream is* UCS09 *or higher.*

3.2.4 Calling party identification

The calling party's geographic location (numbering plan area [NPA]) is vital to some services that may be offered via AIN. As a result, the SSP shall always transmit this information for those queries where calling party identification is expected. In addition, the calling party ID or NPA is used for AIN global title translation (see section B.3).

R3-18 **[21]** The caller's calling line ID (CLID) or ANI shall be transmitted in the `CallingPartyID` parameter if known; otherwise, the caller's NPA shall be transmitted.

R3-19 **[22]** A pseudo-ANI (PANI) shall never be used for the `CallingPartyID` parameter, as it may not contain an NPA in the first three digits.

R3-20 **[320v2]** If inswitch datafill contains a NIL Serving NPA (SNPA) for the trunk, then the default SNPA from inswitch datafill shall be transmitted.

R3-21 **[321v2]** When the originating access is an SS7 trunk, the Nature of Address, Numbering Plan, and Presentation Restriction Indicator received in the IAM shall be transmitted to the SCP.

R3-22 **[814]** When the originating access is a PRI trunk, the Nature of Address and Presentation Restriction Indicator in the SETUP message shall be transmitted to the SCP.

R3-23 **[322v3]** When the originating access is not an SS7 nor a PRI trunk, the Nature of Address sent to the SCP shall be national. When the originating access is not an SS7 trunk, the Numbering Plan sent to the SCP shall be ISDN.

R3-24 **[815]** If the `CallingPartyID` parameter is included in an Analyze_Route message from the SCP, its value shall be transmitted to the SCP via the outgoing `CallingPartyID` parameter in all subsequent interactions between the SSP and SCP for the same call.

3.3 Close message

The `Close` message is sent by the SSP to terminate an open transaction (for example, when terminating EDP processing). This message may also be used as an error message — see Chapter 6 for details — in which case the description of the message is slightly different. It is described in section 4.5.5 of GR-1298-CORE and section 5.2.1.2 of GR-1299-CORE.

R3-25 **[165]** The `Close` message shall be sent by the SSP when the call completes AIN call processing or is given final treatment, and a TCAP transaction with the SCP remains open.

R3-26 **[166]** The `Close` message shall be populated with the parameters shown in Table 3-20.

Table 3-20 Parameters in the Close message

Name	Reference	
	This doc	GR-1299-CORE
UserID	7.49	6.134
BearerCapability	7.13	6.28
CloseCause	7.22	6.45
ExtensionParameter	7.30	6.68

R3-27 **[167]** The `UserID` parameter shall be filled as described in section 3.2.2.

R3-28 **[168]** The `BearerCapability` parameter shall be filled with the bearer capability of the agent as provisioned in the SSP when the `Close` message is sent.

R3-29 **[169]** The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8,

"Extension parameters" for a description of the currently supported private parameters.

R3-30 **[323]** (Deleted)

3.4 Info_Analyzed message

The `Info_Analyzed` message is a TDP-R message sent by the SSP from the `Analyze_Information` PIC. It is described in section 4.5.3.3 of GR-1298-CORE and section 5.2.1.5 of GR-1299-CORE.

R3-31 **[23v3]** The `Info_Analyzed` message shall be sent as a result of the following triggers: `Specific_Feature_Code`, `Customized_Dialing_Plan`, `Specific_Digit_String`, or `Office_Code`.

R3-32 **[24v4]** The `Info_Analyzed` message shall be populated with the parameters shown in Table 3-21.

Table 3-21 Parameters in the Info_Analyzed message

Parameter name	Paragraph reference	
	This doc	GR-1299-CORE
UserID	7.49	6.134
BearerCapability	7.13	6.28
CalledPartyID	7.15	6.33
Lata	7.38	6.89
TriggerCriteriaType	7.48	6.133
ChargeNumber	7.19	6.41
CallingPartyID	7.16	6.36
ChargePartyStationType	7.20	6.42
Carrier	7.17	6.37
AccessCode	7.3	6.1
CollectedAddressInfo	7.23	6.46
CollectedDigits	7.24	6.47
VerticalServiceCode	7.50	6.135
ACGEncountered	N/A	6.2
ExtensionParameter	7.30	6.68
JurisdictionInformation	7.37	6.88

R3-33 **[25]** The `UserID` parameter shall be filled as described in section 3.2.2.

R3-34 **[26v2]** The `BearerCapability` parameter shall be filled with the bearer capability of the agent as provisioned in the SSP when the `Info_Analyzed` message is sent. For messages resulting from the `Office_Code` trigger, the `BearerCapability` parameter shall be restricted to only those values that are allowed for the AIN 0.1 specification.

R3-35 **[27]** The `CalledPartyID` parameter shall be filled with the known address of the called party. Note that this may differ from the collected address as a result of SSP translations.

R3-36 **[816]** The `Lata` parameter shall be filled with the Local Access and Transport Area (LATA) number of the originating facility. *This parameter is supported for this message only if the stream designation of the CAIN protocol version is* UCS11 *or higher.*

R3-37 **[28v2]** (Deleted)

R3-38 **[29v3]** (Deleted)

R3-39 **[452]** (Deleted)

R3-40 **[765]** (Deleted)

R3-41 **[30]** (Deleted)

R3-42 **[324v2]** The `ChargePartyStationType` parameter shall be filled with the Information Digits from the call. This parameter shall be sent in queries resulting from the `Office_Code` trigger *only if parameter* LNP_PARAMETER_SET *in table* CAINPARM *is set to* COMPLETE_SET. This parameter is supported for queries resulting from triggers other than `Office_Code` *only if the stream designation of the CAIN protocol version is* UCS07 *or higher.*

R3-43 **[31v2]** The `Carrier` parameter shall be filled with the Carrier Identification Code (CIC), if known; otherwise, this parameter shall not be transmitted. If the CIC is derived from provisioned data, the Carrier Selection indicator shall be set to “presubscribed and not input.”.

R3-44 **[32v2]** The `AccessCode` parameter shall be filled with the account code, if known. If no account code is identified for the call, the `AccessCode` parameter shall not be transmitted. The `AccessCode` parameter is *only sent if the CAIN protocol version is* V2 *or lower.*

R3-45 **[33]** The `CollectedAddressInfo` parameter shall be filled with the address collected from the incoming agent (for example, via IAM or Setup message, via subscriber dialing, and so forth).

R3-46 **[34v2]** The `CollectedDigits` parameter shall be filled with collected PIN digits. If no PIN code is identified for the call, the

Nortel Networks Proprietary and Confidential

`CollectedDigits` parameter shall not be transmitted. The `CollectedDigits` parameter shall only be sent in the `Info_Analyzed` message *if the CAIN protocol version is* V2 *or lower.*

R3-47 **[170]** The `VerticalServiceCode` parameter shall be filled with the feature code used to invoke a feature (*XX or #XX); if no feature code has been dialed, this parameter shall not be transmitted. *This parameter is supported for this message only if the stream designation of the CAIN protocol version is* UCS06 *or higher.*

R3-48 **[650]** The `ACGEncountered` parameter shall contain the control indicator (SCP or SOCC), the control type, and the number of digits in the control when an ACG control is encountered by a query message and the gap interval has expired or the control has a zero-gap interval.

R3-49 **[35]** The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, “Extension parameters” for a description of the currently supported private parameters.

R3-50 **[453]** If the originator is an SS7 trunk, the `JurisdictionInformation` parameter shall be filled with incoming JIP SS7 parameter if present. *This parameter is supported for this message only if the stream designation of the CAIN protocol version is* UCS08 *or higher.*

R3-51 **[454]** The `JurisdictionInformation` parameter shall be filled with the default JIP provisioned for the originator if the originator is an SS7 trunk but no incoming JIP was received. *This parameter is supported only if the stream designation of the CAIN protocol version is* UCS08 *or higher.*

R3-52 **[455]** The `JurisdictionInformation` parameter shall be filled with the default JIP provisioned for the originator if the originator is a non-SS7 trunk. *This parameter is supported for this message only if the stream designation of the CAIN protocol version is* UCS08 *or higher.*

3.5 Info_Collected message

The `Info_Collected` message is a TDP-R message sent by the SSP from the `Collect_Information` PIC. It is described in section 4.5.3.2 of GR-1298-CORE and section 5.2.1.6 of GR-1299-CORE.

R3-53 **[171v2]** The `Info_Collected` message shall be sent as a result of one of the following triggers: `Off_Hook_Delay`, `PRI_B_Channel` or `Shared_Interoffice_Trunk`.

R3-54 **[172v3]** The `Info_Collected` message shall be populated with the parameters shown in Table 3-22.

Table 3-22 Parameters in the Info_Collected message

Name	Reference	
	This doc	GR-1299-CORE
UserID	7.49	6.134
BearerCapability	7.13	6.28
ChargeNumber	7.19	6.41
Lata	7.38	6.89
Carrier	7.17	6.37
TriggerCriteriaType	7.48	6.133
CallingPartyID	7.16	6.36
ChargePartyStationType	7.20	6.42
AccessCode	7.3	6.1
CollectedAddressInfo	7.23	6.46
CollectedDigits	7.24	6.47
VerticalServiceCode	7.50	6.135
ACGEncountered	N/A	6.2
ExtensionParameter	7.30	6.68

R3-55 **[173]** The `UserID` parameter shall be filled as described in section 3.2.2.

R3-56 **[174]** The `BearerCapability` parameter shall be filled with the bearer capability of the agent as provisioned in the SSP when the `Info_Collected` message is sent.

R3-57 **[175]** (Deleted)

R3-58 **[456]** (Deleted)

R3-59 **[766v2]** (Deleted)

R3-60 **[817]** The `Lata` parameter shall be filled with the Local Access and Transport Area (LATA) number of the originating facility. *This parameter is supported for this message only if the stream designation of the CAIN protocol version is* UCS11 *or higher.*

R3-61 **[176]** The `Carrier` parameter shall be filled with the Carrier Identification Code (CIC), if known; otherwise, this parameter shall not be transmitted. If the CIC is derived from provisioned data, the Carrier Selection indicator shall be set to "presubscribed and not input."

R3-62 **[177v2]** (Deleted)

R3-63 **[178]** The `CallingPartyID` parameter shall be filled as described in section 3.2.4.

R3-64 **[325v2]** The `ChargePartyStationType` parameter shall be filled with the Information Digits from the call. *This parameter is supported for this message only if the stream designation of the CAIN protocol version is* UCS07 *or higher.*

R3-65 **[179v2]** The `AccessCode` parameter shall be filled with the account code, if known. If no account code is identified for the call, the `AccessCode` parameter shall not be transmitted. The `AccessCode` parameter shall only be sent *if the CAIN protocol version is* V2 *or lower.*

R3-66 **[180]** The `CollectedAddressInfo` parameter shall be filled with the address collected from the incoming agent (for example, via IAM or Setup message, or via subscriber dialing, and so forth).

R3-67 **[181v3]** The `CollectedDigits` parameter shall be filled with collected PIN code digits. If no PIN code is identified for the call, the `CollectedDigits` parameter shall not be transmitted. The `CollectedDigits` shall only be sent in the `Info_Collected` message *if the CAIN protocol version is* V2 *or lower.*

R3-68 **[182]** The `VerticalServiceCode` parameter shall be filled with the feature code used to invoke a feature (*XX or #XX); if no feature code has been dialed, this parameter shall not be transmitted.

R3-69 **[651]** The `ACGEncountered` parameter shall contain the control indicator (SCP or SOCC), the control type, and the number of digits in the control when an ACG control is encountered by a query message and the gap interval has expired or the control has a zero-gap interval

R3-70 **[183]** The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, "Extension parameters" for a description of the currently supported private parameters.

3.6 Network_Busy message

The `Network_Busy` message is a TDP-R or EDP-R message sent by the SSP at the `Select_Route` PIC. It is described in section 4.5.3.4 of GR-1298-CORE and section 5.2.1.7 of GR-1299-CORE.

R3-71 **[184]** The `Network_Busy` message shall only be sent as the result of the `Network_Busy` event when the SSP detects that all routes are busy or otherwise unavailable (either locally or as the result of receiving a specific release cause from another switch).

R3-72 **[185v3]** The `Network_Busy` TDP-R message shall be populated with the parameters shown in Table 3-23.

Table 3-23 Parameters in the Network_Busy TDP-R message

Name	Reference	
	This doc	**GR-1299-CORE**
UserID	7.49	6.134
BearerCapability	7.13	6.28
ChargeNumber	7.19	6.41
Lata	7.38	6.89
TriggerCriteriaType	7.48	6.133
CallingPartyID	7.16	6.36
ChargePartyStationType	7.20	6.42
CalledPartyID	7.15	6.33
Carrier	7.17	6.37
ACGEncountered	N/A	6.2
ExtensionParameter	7.30	6.68
BusyCause	7.14	6.31

R3-73 **[326v2]** The `Network_Busy` EDP-R message shall be populated with the parameters shown in Table 3-24.

Table 3-24 Parameters in the Network_Busy EDP-R message (page 1 of 2)

Name	Reference	
	This doc	**GR-1299-CORE**
UserID	7.49	6.134
BearerCapability	7.13	6.28
ExtensionParameter	7.30	6.68

Name	Reference	
	This doc	GR-1299-CORE
NotificationIndicator	N/A	6.94
BusyCause	7.14	6.31

R3-74 **[186]** The `UserID` parameter shall be filled as described in section 3.2.2.

R3-75 **[187]** The `BearerCapability` parameter shall be filled with the bearer capability of the agent as provisioned in the SSP when the `Network_Busy` message is sent.

R3-76 **[188v3]** (Deleted)

R3-77 **[457]** (Deleted)

R3-78 **[767]** (Deleted)

R3-79 **[818]** The `Lata` parameter shall be filled with the Local Access and Transport Area (LATA) number of the originating facility. *This parameter is supported for this message only if the stream designation of the CAIN protocol version is* UCS11 *or higher.*

R3-80 **[189]** The `TriggerCriteriaType` parameter shall be `networkBusy`.

R3-81 **[190]** The `CallingPartyID` parameter shall be filled as described in section 3.2.4.

R3-82 **[327]** The `ChargePartyStationType` parameter shall be filled with the Information Digits from the call. *This parameter is supported for this message only if the stream designation of the CAIN protocol version is* UCS07 *or higher.*

R3-83 **[191]** The `CalledPartyID` parameter shall be filled with the known address of the called party. Note that this may differ from the collected address as a result of SSP translations.

R3-84 **[192]** The `Carrier` parameter shall be filled with the Carrier Identification Code (CIC), if known; otherwise, this parameter shall not be transmitted. If the CIC is derived from provisioned data, the Carrier Selection indicator shall be set to “presubscribed and not input.”

R3-85 **[652]** The `ACGEncountered` parameter shall contain the control indicator (SCP or SOCC), the control type, and the number of digits in the control when an ACG control is encountered by a query message and the gap interval has expired or the control has a zero-gap interval

R3-86 **[193]** The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8,

"Extension parameters" for a description of the currently supported private parameters.

R3-87 **[328]** The `NotificationIndicator` parameter shall identify the `Network_Busy` message as a Switch Request Message.

R3-88 **[194v3]** The `BusyCause` parameter shall contain the network-specific cause value received from a PRI or ISUP agent when releasing the call. This parameter shall contain the network-specific cause value `no_circuit_available` for DAL and AXXESS agents when releasing the call. This parameter shall be supported by NetworkBuilder for the `Network_Busy` message, although GR-1299-CORE does not support the `BusyCause` parameter in the `Network_Busy` message. *In the EDP-R message, this parameter is supported for this message only if the value of RESTRICT_NETBUSY_BUSYCAUSE in table CAINPARM is N.*

3.7 Origination_Attempt message

The `Origination_Attempt` message is a TDP-R message sent by the SSP from the `O_Null` PIC. It is described in section 4.5.3.1 of GR-1298-CORE and section 5.2.1.16 of GR-1299-CORE.

R3-89 **[195]** The `Origination_Attempt` message shall only be sent from the `Off_Hook_Immediate` trigger as applied to private trunks.

R3-90 **[329]** The `Origination_Attempt` message is never sent as an EDP-R message, although GR-1299-CORE supports the `Origination_Attempt` message as an EDP-R message.

R3-91 **[196v3]** The `Origination_Attempt` message shall be populated with the parameters shown in Table 3-25.

Table 3-25 Parameters in the Origination_Attempt message (page 1 of 2)

Name	Reference	
	This doc	GR-1299-CORE
UserID	7.49	6.134
BearerCapability	7.13	6.28
ChargeNumber	7.19	6.41
Lata	7.38	6.89
TriggerCriteriaType	7.48	6.133

Name	Reference	
	This doc	GR-1299-CORE
CallingPartyID	7.16	6.36
Carrier	7.17	6.37
ACGEncountered	N/A	6.2
ExtensionParameter	7.30	6.68

R3-92 **[197]** The `UserID` parameter shall be filled as described in section 3.2.2.

R3-93 **[198]** The `BearerCapability` parameter shall be filled with the bearer capability of the agent as provisioned in the SSP when the `Origination_Attempt` message is sent.

R3-94 **[199]** (Deleted)

R3-95 **[458]** (Deleted)

R3-96 **[768]** (Deleted)

R3-97 **[819]** The `Lata` parameter shall be filled with the Local Access and Transport Area (LATA) number of the originating facility. *This parameter is supported for this message only if the stream designation of the CAIN protocol version is* UCS11 *or higher.*

R3-98 **[200]** The `TriggerCriteriaType` parameter shall be `offHookImmediate`.

R3-99 **[201]** The `CallingPartyID` parameter shall be filled as described in section 3.2.4.

R3-100 **[202]** The `Carrier` parameter shall be filled with the Carrier Identification Code (CIC), if present in switch datafill for the originating trunk; otherwise, this parameter shall not be transmitted. The Carrier Selection indicator shall always be set to “presubscribed and not input”.

R3-101 **[653]** The `ACGEncountered` parameter shall contain the control indicator (SCP or SOCC), the control type, and the number of digits in the control when an ACG control is encountered by a query message and the gap interval has expired or the control has a zero-gap interval

R3-102 **[203]** The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, “Extension parameters” for a description of the currently supported private parameters.

3.8 O_Answer message

The O_Answer message is an EDP-N message sent by the SSP at the O_Alerting PIC. It is described in section 4.5.3.7 of GR-1298-CORE and section 5.2.1.9 of GR-1299-CORE.

R3-103 [330] The O_Answer message shall be sent as the result of the O_Answer requested event, when the terminating party answers the call.

R3-104 [331v2] The O_Answer message shall be populated with the following parameters shown in Table 3-26.

Table 3-26 Parameters in the O_Answer message

Name	Reference	
	This doc	GR-1299-CORE
UserID	7.49	6.134
BearerCapability	7.13	6.28
NotificationIndicator	N/A	6.94
ExtensionParameter	7.30	6.68

R3-105 [332] The UserID parameter shall be filled as described in section 3.2.2.

R3-106 [333] The BearerCapability parameter shall be filled with the bearer capability of the agent as provisioned in the SSP when the O_Answer message is sent.

R3-107 [334] The NotificationIndicator parameter shall identify the O_Answer message as a Switch Notification Message.

R3-108 [335] The ExtensionParameter, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, “Extension parameters” for a description of the currently supported private parameters.

3.9 O_Called_Party_Busy message

The O_Called_Party_Busy message is a TDP-R or EDP-R message sent by the SSP at the Send_Call and O_Alerting PICs. For Carrier AIN, this message is supported only at the Send_Call PIC. It is described in section 4.5.3.6 of GR-1298-CORE and section 5.2.1.10 of GR-1299-CORE.

R3-109 [204] The O_Called_Party_Busy message shall be sent as the result of the O_Called_Party_Busy event, when the SSP detects user busy on the terminating SS7, PRI, or on-net DAL agent.

R3-110 [205v3] The `O_Called_Party_Busy` TDP-R message shall be populated with the following parameters shown in Table 3-27.

Table 3-27 Parameters in the O_Called_Party_Busy TDP-R message

Name	Reference	
	This doc	GR-1299-CORE
UserID	7.49	6.134
BearerCapability	7.13	6.28
CalledPartyID	7.15	6.33
Lata	7.38	6.89
TriggerCriteriaType	7.48	6.133
ChargeNumber	7.19	6.41
CallingPartyID	7.16	6.36
ChargePartyStationType	7.20	6.42
Carrier	7.17	6.37
BusyCause	7.14	6.31
ACGEncountered	N/A	6.2
ExtensionParameter	7.30	6.68

R3-111 [336v2] The `O_Called_Party_Busy` EDP-R message shall be populated with the following parameters shown in Table 3-28.

Table 3-28 Parameters in the O_Called_Party_Busy EDP-R message

Name	Reference	
	This doc	GR-1299-CORE
UserID	7.49	6.134
BearerCapability	7.13	6.28
BusyCause	7.14	6.31
NotificationIndicator	N/A	6.94
ExtensionParameter	7.30	6.68

R3-112 [206] The `UserID` parameter shall be filled as described in section 3.2.2.

R3-113 [207] The `BearerCapability` parameter shall be filled with the bearer capability of the agent as provisioned in the SSP when the `O_Called_Party_Busy` message is sent.

R3-114 [208] (Deleted)

R3-115 [820] The `Lata` parameter shall be filled with the Local Access and Transport Area (LATA) number of the originating facility. *This parameter is supported for this message only if the stream designation of the CAIN protocol version is* UCS11 *or higher.*

R3-116 [209] The `TriggerCriteriaType` parameter shall be `oCalledPartyBusy`.

R3-117 [459] (Deleted)

R3-118 [769] (Deleted)

R3-119 [210] The `CallingPartyID` parameter shall be filled as described in section 3.2.4.

R3-120 [211] The `CalledPartyID` parameter shall be filled with the known address of the called party. Note that this may differ from the collected address as a result of SSP translations.

R3-121 [337] The `ChargePartyStationType` parameter shall be filled with the Information Digits from the call. *This parameter is supported for this message only if the stream designation of the CAIN protocol version is* UCS07 *or higher.*

R3-122 [212] The `Carrier` parameter shall be filled with the Carrier Identification Code (CIC), if known; otherwise, this parameter shall not be transmitted. If the CIC is derived from provisioned data, the Carrier Selection indicator shall be set to "presubscribed and not input."

R3-123 [213v2] The `BusyCause` parameter shall contain the network-specific cause value received from a PRI or ISUP agent when releasing the call. This parameter shall contain the network-specific cause value `user_busy` for DAL and AXXESS agents when releasing the call.

R3-124 [338] The `NotificationIndicator` parameter shall identify the `O_Called_Party_Busy` message as a Switch Request Message.

R3-125 [654] The `ACGEncountered` parameter shall contain the control indicator (SCP or SOCC), the control type, and the number of digits in the control when an ACG control is encountered by a query message and the gap interval has expired or the control has a zero-gap interval

R3-126 [214v2] (Deleted)

R3-127 [215] The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8,

"Extension parameters" for a description of the currently supported private parameters.

3.10 O_Feature_Requested message

The `O_Feature_Requested` message is a TDP-R message sent by the SSP at any of the `O_Null`, `Collect_Information`, `Analyze_Information`, `Select_Route`, and `Authorize_Call_Setup` PICs. For Carrier AIN, it is supported only when sent from the `Collect_Information` PIC. It is not described in either GR-1298-CORE or GR-1299-CORE.

This message is not part of the Bellcore AIN specification (GR-1299-CORE). Encoding for this message is described in section A.3.

R3-128 [216] The `O_Feature_Requested` message shall be sent as the result of the `O_Feature_Requested` trigger.

R3-129 [217v3] The `O_Feature_Requested` message shall be populated with the parameters shown in Table 3-29.

Table 3-29 Parameters in the O_Feature_Requested message (page 1 of 2)

Name	Reference	
	This doc	GR-1299-CORE
UserID	7.49	6.134
BearerCapability	7.13	6.28
ChargeNumber	7.19	6.41
Lata	7.38	6.89
Carrier	7.17	6.37
TriggerCriteriaType	7.48	6.133
PointInCall	7.41	6.101
FeatureActivatorID	7.32	6.75
CallingPartyID	7.16	6.36
ChargePartyStationType	7.20	6.42
AccessCode	7.3	6.1
CollectedAddressInfo	7.23	6.46
CollectedDigits	7.24	6.47
VerticalServiceCode	7.50	6.135

Name	Reference	
	This doc	GR-1299-CORE
ACGEncountered	N/A	6.2
ExtensionParameter	7.30	6.68

R3-130 [218] The `UserID` parameter shall be filled as described in section 3.2.2.

R3-131 [219] The `BearerCapability` parameter shall be filled with the bearer capability of the agent as provisioned in the SSP when the `O_Feature_Requested` message is sent.

R3-132 [220] (Deleted)

R3-133 [460] (Deleted)

R3-134 [770] (Deleted)

R3-135 [821] The `Lata` parameter shall be filled with the Local Access and Transport Area (LATA) number of the originating facility. *This parameter is supported for this message only if the stream designation of the CAIN protocol version is* UCS11 *or higher.*

R3-136 [221] The `Carrier` parameter shall be filled with the Carrier Identification Code (CIC), if known; otherwise, this parameter shall not be transmitted. If the CIC is derived from provisioned data, the Carrier Selection indicator shall be set to “presubscribed and not input.”

R3-137 [222] The `TriggerCriteriaType` parameter shall be `oFeatureActivator`.

R3-138 [223] The `PointInCall` parameter shall indicate the PIC at which the trigger was checked.

R3-139 [224v2] The `FeatureActivatorID` parameter shall contain the number of the feature from the trigger table which caused the query to be sent.

R3-140 [225] The `CallingPartyID` parameter shall be filled as described in section 3.2.4.

R3-141 [339] The `ChargePartyStationType` parameter shall be filled with the Information Digits from the call. *This parameter is supported for this message only if the stream designation of the CAIN protocol version is* UCS07 *or higher.*

R3-142 [226v2] The `AccessCode` parameter shall be filled with the account code, if known. If no account code is identified for the call, the `AccessCode` parameter shall not be transmitted. The `AccessCode` parameter is only sent *if the CAIN protocol version is* V2 *or lower.*

R3-143 [227] The `CollectedAddressInfo` parameter shall be filled with the address collected from the incoming agent (for example, via IAM or Setup message, or via subscriber dialing, and so forth).

R3-144 [228v2] The `CollectedDigits` parameter shall be filled with collected PIN code digits. If no PIN code is identified for the call, the `CollectedDigits` parameter shall not be transmitted. The `CollectedDigits` parameter is only sent in the O_Feature_Requested message *if the CAIN protocol version is* V2 *or lower.*

R3-145 [229] The `VerticalServiceCode` parameter shall be filled with the feature code used to invoke a feature (*XX or #XX); if no feature code has been dialed, this parameter shall not be transmitted.

R3-146 [655] The `ACGEncountered` parameter shall contain the control indicator (SCP or SOCC), the control type, and the number of digits in the control when an ACG control is encountered by a query message and the gap interval has expired or the control has a zero-gap interval

R3-147 [230] The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, "Extension parameters" for a description of the currently supported private parameters.

3.11 O_No_Answer message

The `O_No_Answer` message is a TDP-R or EDP-R message sent by the SSP at the `Send_Call` and `O_Alerting` PICs. For Carrier AIN, this message is supported only at the `O_Alerting` PIC. It is described in section 4.5.3.8 of GR-1298-CORE and section 5.2.1.13 of GR-1299-CORE.

R3-148 [231] The `O_No_Answer` message shall be sent as the result of the `O_No_Answer` event, when the SSP fails to detect answer prior to the expiration of the originating No Answer timer.

R3-149 [232v3] The `O_No_Answer` TDP-R message shall be populated with the parameters shown in Table 3-30.

Table 3-30 Parameters in the O_No_Answer TDP-R message (page 1 of 2)

Name	Reference	
	This doc	GR-1299-CORE
UserID	7.49	6.134
BearerCapability	7.13	6.28
CalledPartyID	7.15	6.33
Lata	7.38	6.89

Name	Reference	
	This doc	GR-1299-CORE
TriggerCriteriaType	7.48	6.133
ChargeNumber	7.19	6.41
CallingPartyID	7.16	6.36
ChargePartyStationType	7.20	6.42
Carrier	7.17	6.37
ACGEncountered	N/A	6.2
ExtensionParameter	7.30	6.68

R3-150 [856] The `O_No_Answer` EDP-R message shall be populated with the parameters shown in Table 3-31.

Table 3-31 Parameters in the O_No_Answer EDP-R message

Name	Reference	
	This doc	GR-1299-CORE
UserID	7.49	6.134
BearerCapability	7.13	6.28
NotificationIndicator	N/A	6.94
ExtensionParameter	7.30	6.68

R3-151 [233] The `UserID` parameter shall be filled as described in section 3.2.2.

R3-152 [234] The `BearerCapability` parameter shall be filled with the bearer capability of the agent as provisioned in the SSP when the `O_No_Answer` message is sent.

R3-153 [235] The `CalledPartyID` parameter shall be filled with the known address of the called party. Note that this may differ from the collected address as a result of SSP translations.

R3-154 [822] The `Lata` parameter shall be filled with the Local Access and Transport Area (LATA) number of the originating facility. *This parameter is supported for this message only if the stream designation of the CAIN protocol version is* UCS11 *or higher.*

R3-155 [236] The `TriggerCriteriaType` parameter shall be `oNoAnswer`.

R3-156 [237] (Deleted)

R3-157 [461] (Deleted)

R3-158 [771] (Deleted)

R3-159 [238] The `CallingPartyID` parameter shall be filled as described in section 3.2.4.

R3-160 [340] The `ChargePartyStationType` parameter shall be filled with the Information Digits from the call. *This parameter is supported for this message only if the stream designation of the CAIN protocol version is* UCS07 *or higher.*

R3-161 [239] The `Carrier` parameter shall be filled with the Carrier Identification Code (CIC), if known; otherwise, this parameter shall not be transmitted. If the CIC is derived from provisioned data, the Carrier Selection indicator shall be set to "presubscribed and not input."

R3-162 [341] The `NotificationIndicator` parameter shall identify the `O_No_Answer` message as a Switch Request Message.

R3-163 [656] The `ACGEncountered` parameter shall contain the control indicator (SCP or SOCC), the control type, and the number of digits in the control when an ACG control is encountered by a query message and the gap interval has expired or the control has a zero-gap interval

R3-164 [240v2] (Deleted)

R3-165 [241] The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, "Extension parameters" for a description of the currently supported private parameters.

3.12 O_Term_Seized message

The `O_Term_Seized` message is an EDP-N message sent by the SSP at the `Send_Call` PIC. It is described in section 4.5.3.5 of GR-1298-CORE and section 5.2.1.15 of GR-1299-CORE.

R3-166 [342] The `O_Term_Seized` message shall be sent as the result of the `O_Term_Seized` requested event, when the terminating party is alerted of the call and audible ringing is provided to the calling party.

R3-167 [343v2] The `O_Term_Seized` message shall be populated with the following parameters shown in Table 3-32.

Table 3-32 Parameters in the O_Term_Seized message

Name	Reference	
	This doc	**GR-1299-CORE**
UserID	7.49	6.134
BearerCapability	7.13	6.28
NotificationIndicator	N/A	6.94
ExtensionParameter	7.30	6.68

R3-168 [344] The `UserID` parameter shall be filled as described in section 3.2.2.

R3-169 [345] The `BearerCapability` parameter shall be filled with the bearer capability of the agent as provisioned in the SSP when the `O_Term_Seized` message is sent.

R3-170 [346] The `NotificationIndicator` parameter shall identify the `O_Term_Seized` message as a Switch Notification Message.

R3-171 [347] The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, "Extension parameters" for a description of the currently supported private parameters.

3.13 O_Mid_Call message

The `O_Mid_Call` message may be sent as a TDP-R or EDP-R message. The TDP-R message is sent by the SSP from the `Send_Call`, `O_Alerting`, `O_Active`, or `O_Suspended` PICs. The EDP-R message is sent by the SSP from the `Send_Call`, `O_Alerting`, or `O_Active` PICs. The `O_Mid_Call` message is described in section 21.4.6 of GR-1298-CORE and section 5.2.1.12 of GR-1299-CORE.

R3-172 [462] The `O_Mid_Call` TDP-R message shall only be sent from `O_IEC_Reorigination` trigger.

R3-173 [463v2] The `O_Mid_Call` TDP-R message shall be populated with the parameters shown in Table 3-33.

Table 3-33 Parameters in the O_Mid_Call TDP-R message

Name	Reference	
	This doc	**GR-1299-CORE**
UserID	7.49	6.134
BearerCapability	7.13	6.28
TriggerCriteriaType	7.48	6.133
LegID	7.39	6.90
PointInCall	7.41	6.101
CcID	7.18	6.40
ChargeNumber	7.19	6.41
CallingPartyID	7.16	6.36
ChargePartyStationType	7.20	6.42
Carrier	7.17	6.37
AccessCode	7.36	6.1
CollectedAddressInfo	7.23	6.46
VerticalServiceCode	7.50	6.135
ACGEncountered	N/A	6.2
ExtensionParameter	7.30	6.68

R3-174 [462] The `O_Mid_Call` EDP-R message shall only be sent upon detecting the Switch_Hook_Flash event.

R3-175 [463] The `O_Mid_Call` EDP-R message shall be populated with the parameters shown in Table 3-34.

Table 3-34 Parameters in the O_Mid_Call EDP-R message (page 1 of 2)

Name	Reference	
	This doc	**GR-1299-CORE**
UserID	7.49	6.134
BearerCapability	7.13	6.28

Name	Reference	
	This doc	GR-1299-CORE
CcID	7.18	6.40
NotificationIndicator	N/A	6.94
ExtensionParameter	7.30	6.68

R3-176 [464] The `UserID` parameter shall be filled as described in section 3.2.2.

R3-177 [465] The `BearerCapability` parameter shall be filled with the bearer capability of the agent as provisioned in the SSP when the `O_Mid_Call` message is sent.

R3-178 [466] The `TriggerCriteriaType` parameter shall contain `oIECReorigination`.

R3-179 [467] The `LegID` parameter shall contain the value 0.

R3-180 [468] The `PointInCall` parameter shall contain one of the following values: `sendCall`, `oAlerting`, `oActive`, or `oSuspended`.

R3-181 [469v3] The `CcID` parameter shall be filled with the current call configuration.

R3-182 [470] (Deleted)

R3-183 [471] (Deleted)

R3-184 [772] (Deleted)

R3-185 [472] The `CallingPartyID` parameter shall be filled as described in section 3.2.4.

R3-186 [473] The `ChargePartyStationType` parameter shall be filled with the Information Digits from the call.

R3-187 [474] The `Carrier` parameter shall be filled with the Carrier Identification Code (CIC), if known; otherwise, this parameter shall not be transmitted. If the CIC is derived from provisioned data, the Carrier Selection indicator shall be set to “presubscribed and not input.”

R3-188 [475] The `AccessCode` parameter shall be filled with the account code, if known. If no account code is identified for the call, the `AccessCode` parameter shall not be transmitted. The `AccessCode` shall only be sent in `O_Mid_Call` messages *if the CAIN protocol version is* V2 *or lower.*

Nortel Networks Proprietary and Confidential

R3-189 [476] The `CollectedAddressInfo` parameter shall be filled with the address collected from the incoming agent (for example, via IAM or Setup message, or via subscriber dialing, and so forth).

R3-190 [477] The `VerticalServiceCode` parameter shall be filled with the feature code used to invoke a feature (*XX or #XX); if no feature code has been dialed, this parameter shall not be transmitted.

R3-191 [657] The `ACGEncountered` parameter shall contain the control indicator (SCP or SOCC), the control type, and the number of digits in the control when an ACG control is encountered by a query message and the gap interval has expired or the control has a zero-gap interval

R3-192 [478] The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, “Extension parameters” for a description of the currently supported private parameters.

R3-193 [777] The `NotificationIndicator` parameter shall identify the `O_Mid_Call` message as a Switch Request Message.

3.14 Termination_Attempt message

The `Termination_Attempt` message is a TDP-R message sent by the SSP at the `T_Null` PIC. It is described in section 4.5.4.1 of GR-1298-CORE and section 5.2.1.24 of GR-1299-CORE.

R3-194 [479] The `Termination_Attempt` message shall only be sent as the result of the `Termination_Attempt` trigger.

R3-195 [480v3] The `Termination_Attempt` message shall be populated with the parameters shown in Table 3-35.

Table 3-35 Parameters in the Termination_Attempt TDP-R message (page 1 of 2)

Name	Reference	
	This doc	GR-1299-CORE
UserID	7.49	6.134
BearerCapability	7.13	6.28
CalledPartyID	7.15	6.33
Lata	7.38	6.89
TriggerCriteriaType	7.48	6.133
ChargeNumber	7.19	6.41
CallingPartyID	7.16	6.36
ChargePartyStationType	7.20	6.42

Name	Reference	
	This doc	GR-1299-CORE
ACGEncountered	N/A	6.2
ExtensionParameter	7.30	6.68

R3-196 [481] The `UserID` parameter shall be filled as described in section 3.2.2 for the terminator.

R3-197 [482] The `BearerCapability` parameter shall be filled with the bearer capability of the originating agent as provision in the SSP when the `Termination_Attempt` message is sent.

R3-198 [483] The `CalledPartyID` parameter shall be filled with the known address of the called party. Note that this may differ from the collected address as a result of SSP translations.

R3-199 [484] (Deleted)

R3-200 [485] (Deleted)

R3-201 [773] (Deleted)

R3-202 [486] (Deleted)

R3-203 [823] The `Lata` parameter shall be filled with the Local Access and Transport Area (LATA) number of the terminating facility. *This parameter is supported for this message only if the stream designation of the CAIN protocol version is* UCS11 *or higher.*

R3-204 [487] The `TriggerCriteriaType` parameter shall contain `terminationAttempt`.

R3-205 [488] The `CallingPartyID` parameter shall be filled as described in section 3.2.4.

R3-206 [489] The `ChargePartyStationType` parameter shall be filled with the Information Digits from the call.

R3-207 [658] The `ACGEncountered` parameter shall contain the control indicator (SCP or SOCC), the control type, and the number of digits in the control when an ACG control is encountered by a query message and the gap interval has expired or the control has a zero-gap interval

R3-208 [490] The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, “Extension parameters” for a description of the currently supported private parameters.

3.15 O_Disconnect message

The `O_Disconnect` message is sent by the SSP at the `O_Active` or `O_Suspended` PICs and may be either an EDP-N or EDP-R message. It is described in section 4.5.3.10 of GR-1298-CORE and section 5.2.1.11 of GR-1299-CORE.

R3-209 [491] The `O_Disconnect` message shall be sent as the result of the `O_Disconnect` requested event.

R3-210 [492v2] The `O_Disconnect` message shall be populated with the parameters shown in Table 3-36.

Table 3-36 Parameters in the O_Disconnect EDP message

Name	Reference	
	This doc	GR-1299-CORE
UserID	7.49	6.134
BearerCapability	7.13	6.28
NotificationIndicator	N/A	6.94
ExtensionParameter	7.30	6.68
LegID	7.39	6.90
PointInCall	7.41	6.101

R3-211 [493] The `UserID` parameter shall be filled as described in section 3.2.2.

R3-212 [494] The `BearerCapability` parameter shall be filled with the bearer capability of the agent as provisioned in the SSP when the `O_Disconnect` message is sent.

R3-213 [495v2] The `NotificationIndicator` parameter shall identify the `O_Disconnect` message as either a Switch Notification Message or a Switch Request Message.

R3-214 [496v2] The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. It is only supported for EDP-N messages. See Chapter 8, “Extension parameters” for a description of the currently supported private parameters.

R3-215 [497v2] The `LegID` parameter shall indicate which leg of the call disconnected, either 0, 1, or 2.

R3-216 [498] The `PointInCall` parameter shall contain either the value `oActive` or `oSuspended`.

3.16 Timeout message

The `Timeout` message is an EDP-R message sent by the SSP at the `O_Active` or `O_Suspended` PICs. It is described in section 21.4.8.1 of GR-1298-CORE and section 5.2.1.26 of GR-1299-CORE.

R3-217 [499] The `Timeout` message shall be sent as the result of the `Timeout` requested event.

R3-218 [500v2] The `Timeout` message shall be populated with the following parameters shown in Table 3-37.

Table 3-37 Parameters in the Timeout EDP-R message

Name	Reference	
	This doc	GR-1299-CORE
UserID	7.49	6.134
BearerCapability	7.13	6.28
CcID	7.18	6.40
ExtensionParameter	7.30	6.68
NotificationIndicator	N/A	6.94

R3-219 [501] The `UserID` parameter shall be filled as described in section 3.2.2.

R3-220 [502] The `BearerCapability` parameter shall be filled with the bearer capability of the agent as provisioned in the SSP when the `Timeout` message is sent.

R3-221 [503v2] The `CcID` parameter shall be filled with the following value: `stable2Party`.

R3-222 [504] The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, “Extension parameters” for a description of the currently supported private parameters.

R3-223 [505] The `NotificationIndicator` parameter shall identify the `Timeout` message as a Switch Request Message.

3.17 Call_Info_From_Resource message

The `Call_Info_From_Resource` message is sent by the SSP to send intermediate information from an IP to the SCP during an 1129-style STR Connection. It is described in section 5.2.1.1 of GR-1299-CORE and section 4 of GR-1129-CORE.

R3-224 [506] The `Call_Info_From_Resource` message shall be sent as an intermediate transport of information between an IP and the SCP during an 1129-style STR Connection.

R3-225 [507] The `Call_Info_From_Resource` message shall be populated with the parameters shown in Table 3-38.

Table 3-38 Parameters in the Call_Info_From_Resource message

Name	Reference	
	This doc	GR-1299-CORE
IPReturnBlock	7.36	6.86
ExtensionParameter	7.30	6.68

R3-226 [508] The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, “Extension parameters” for a description of the currently supported private parameters.

R3-227 [509] The `Call_Info_From_Resource` is correlated to the `Send_To_Resource` or `Connect_To_Resource` which instructed the SSP to begin the 1129-style STR Connection.

3.18 Resource_Clear message

The `Resource_Clear` message is sent by the SSP in response to a conversational `Send_To_Resource` message from the SCP. It is described in section 5.5 of GR-1298-CORE and section 5.2.1.17 of GR-1299-CORE.

R3-228 [242] The SSP shall always send the `Resource_Clear` message as the response to a `Send_To_Resource` message transmitted in a Conversation package, unless a TCAP error occurs (see Chapter 6).

R3-229 [243] If the `Resource_Clear` message is sent to indicate caller abandon of the call (except as noted below), or as the result of an error condition, it shall be sent in a Response package, and the TCAP transaction shall be not be maintained.

R3-230 [244] (Deleted)

R3-231 [245] In all other cases, the `Resource_Clear` message shall be sent in a Conversation package, and the SSP shall maintain the transaction.

R3-232 [246] The `Resource_Clear` message shall be correlated to a `Send_To_Resource` message unless the caller interaction is terminated by the receipt of a `Cancel_Resource_Event` message, in which case the `Resource_Clear` shall be correlated to

the `Cancel_Resource_Event` and the SSP shall not respond to the `Send_To_Resource` message.

R3-233 [247v3] The `Resource_Clear` message shall be populated with the parameters shown in Table 3-39.

Table 3-39 Parameters in the Resource_Clear message

Name	Reference	
	This doc	GR-1299-CORE
ClearCause	7.21	6.43
CollectedDigits	7.24	6.47
CollectedAddressInfo	7.23	6.46
FailureCause	7.31	6.73
ClearCauseData	N/A	6.44
IPReturnBlock	7.36	6.86
ExtensionParameter	7.30	6.68

R3-234 [248] (Deleted)

R3-235 [249] (Deleted)

R3-236 [250] The `CollectedDigits` parameter shall be included only if the `Resource_Clear` message is in response to a `Send_To_Resource` message that requested digit collection by the SSP.

R3-237 [251] The `CollectedDigits` parameter is an undifferentiated sequence of digits collected by the SSP. The SCP shall be responsible for separating these digits into numbers (for example, account code and PIN digits).

R3-238 [659] The `CollectedAddressInfo` parameter is an address collected by the SSP in response to a request by the SCP for a “normal” number of address digits to be collected based on the dialling plan in force at the time of the collection of the address. (GR-1298-CORE [469]).

R3-239 [252] The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, “Extension parameters” for a description of the currently supported private parameters.

3.18.1 Intelligent Peripheral Requirements

R3-240 [510] The `ClearCauseData` parameter shall be included only if the `Resource_Clear` message is in response to a `Send_To_Resource` message that requested the call be connected to an IP and the IP provides additional error information in a Return Error component to the SSP and the error value in the Return Error component is `taskRefused`. The additional error information shall be unmodified and copied into the `ClearCauseData` parameter.

R3-241 [511] The `ClearCauseData` parameter shall only be sent when a universal octet string tag (#04) is present in the additional information in the Return Error component.

R3-242 [512] The `IPReturnBlock` parameter shall be included if the `Resource_Clear` message is in response to a `Send_To_Resource` message which contains a `DestinationAddress` parameter and the IP sends the SSP an ISDN or SS7 message containing a Return Result component with the `IPReturnBlock` parameter. The incoming `IPReturnBlock` parameter shall be sent in the `IPReturnBlock` parameter of the `Resource_Clear` message.

R3-243 [513] The `IPReturnBlock` parameter shall be included if the `Resource_Clear` message is in response to a `Send_To_Resource` message without the `DestinationAddress` and with a `STRParameterBlock` encoded as a `FlexParameterBlock` with a Nortel Networks encoding authority.

R3-244 [514] The `IPReturnBlock` shall only be included if **[512]** or **[513]** are satisfied.

3.19 CTR_Clear message

The `CTR_Clear` message is sent by the SSP in response to a conversation `Connect_To_Resource` message for the SCP. It is described in section 21.6.2.6 of GR-1298-CORE and section 5.2.1.3 of GR-1299-CORE.

R3-245 [514] The SSP shall always send the `CTR_Clear` message as the response to a `Connect_To_Resource` message.

R3-246 [515] If the `CTR_Clear` message is sent to indicate caller abandon of the call, or as the result of an error condition, it shall be sent in a Response package, and the TCAP transaction shall not be maintained.

R3-247 [516] In all other cases, the `CTR_Clear` message shall be sent in a Conversation package, and the SSP shall maintain the transaction.

R3-248 [517] The `CTR_Clear` message shall be correlated to a `Connect_To_Resource` message.

R3-249 [518] The `CTR_Clear` message shall be populated with the parameters shown in Table 3-40.

Table 3-40 Parameters in the CTR_Clear message

Name	Reference	
	This doc	GR-1299-CORE
ClearCause	7.21	6.43
LegID	7.39	6.90
CcID	7.18	6.40
CollectedDigits	7.24	6.47
CollectedAddressInfo	7.23	6.46
FailureCause	7.31	6.73
ClearCauseData	N/A	6.44
IPReturnBlock	7.36	6.86
ExtensionParameter	7.30	6.68

R3-250 [519] If the `CTR_Clear` message indicates caller abandon, the `ClearCause` parameter shall be set to `userAbandon`.

R3-251 [520v3] The `LegID` parameter shall indicate which leg was involved in the interaction, either 0, 1 or 2.

R3-252 [521v2] The `CcID` parameter shall be `originatingSetup` or `stable2party`.

R3-253 [522] The `CollectedDigits` parameter shall be included only if the `CTR_Clear` message is in response to a `Connect_to_Resource` message that requested digit collection by the SSP.

R3-254 [660] The `CollectedAddressInfo` parameter is an address collected by the SSP in response to a request by the SCP for a “normal” number of address digits to be collected based on the dialling plan in force at the time of the collection of the address. (GR-1298-CORE [469]).

3.19.1 Intelligent Peripheral Requirements

R3-255 [523] The `IPReturnBlock` parameter shall be included if the `CTR_Clear` message is in response to a `Connect_To_Resource` message which contains a `DestinationAddress` parameter and the IP sends the SSP an ISDN or SS7 message containing a Return

Result component with the IPReturnBlock parameter. The incoming IPReturnBlock parameter shall be unmodified and sent in the IPReturnBlock parameter of the CTR_Clear message.

R3-256 [524] The IPReturnBlock parameter shall be included if the CTR_Clear message is in response to a Connect_To_Resource message without the DestinationAddress and with a STRParameterBlock encoded as a FlexParameterBlock with a Nortel Networks encoding authority.

R3-257 [525] The IPReturnBlock shall only be included if **[523]** or **[524]** are satisfied.

3.20 O_Abandon message

The O_Abandon message is an EDP-R message sent by the SSP at the Collect_Information, Analyze_Information, Select_Route, Send_Call, or O_Alerting PICs. It is described in section 21.4.9.1 of GR-1298-CORE and section 5.2.1.8 of GR-1299-CORE.

R3-258 [779] The O_Abandon EDP-R message shall be sent as the result of the O_Abandon requested event.

R3-259 [780] The O_Abandon EDP-R message shall be populated with the parameters shown in Table 3-41.

Table 3-41 Parameters in the O_Abandon EDP-R message

Name	Reference	
	This doc	GR-1299-CORE
UserID	7.49	6.134
BearerCapability	7.13	6.28
CcID	7.18	6.40
PointInCall	7.37	6.101
ExtensionParameter	7.30	6.68
NotificationIndicator	N/A	6.94

R3-260 [781] The UserID parameter shall be filled as described in section 3.2.2.

R3-261 [782] The BearerCapability parameter shall be filled with the bearer capability of the agent as provisioned in the SSP when the O_Abandon message is sent.

R3-262 [783] The CcID parameter shall be filled with the current call configuration.

R3-263 [784] The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, "Extension parameters" for a description of the currently supported private parameters.

R3-264 [785] The `NotificationIndicator` parameter shall identify the `O_Abandon` message as a Switch Request Message.

R3-265 [786] The `PointInCall` parameter shall contain one of the following values: `Collect_Information`, `Analyze_Information`, `Select_Route`, `Send_Call`, or `O_Alerting`.

3.21 Failure_Outcome message

The `Failure_Outcome` message is sent by the SSP to inform the SCP that it was unable to perform a requested operation. It is described in section 21.4.13.5 of GR-1298-CORE and section 5.2.1.4 of GR-1299-CORE.

R3-266 [813] The `Failure_Outcome` message shall be populated with the parameters shown in Table 3-42.

Table 3-42 Parameters in the Failure_Outcome message

Name	Reference	
	This doc	GR-1299-CORE
UserID	7.49	6.134
BearerCapability	7.13	6.28
CcID	7.18	6.40
LegID	7.39	6.90
PointInCall	7.37	6.101
FailureCause	7.31	6.73
ExtensionParameter	7.30	6.68

R3-267 [837] The `UserID` parameter shall be filled as described in section 3.2.2.

R3-268 [838] The `BearerCapability` parameter shall be filled with the bearer capability of the agent as provisioned in the SSP when the `Failure_Outcome` message is sent.

R3-269 [839] The `CcID` parameter shall be filled with the current call configuration.

R3-270 [840] The `LegID` parameter shall contain a value of either 0, 1, or 2, depending on the nature of the failure.

R3-271 [841] The `PointInCall` parameter shall be filled with the current point in call.

R3-272 [842] The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, “Extension parameters” for a description of the currently supported private parameters.

Chapter 4: SCP call-related messages

4.1 Introduction

This chapter provides an overview of the service control point (SCP) call-related messages supported by the NetworkBuilder implementation of Carrier AIN, together with rules for populating the messages with parameters. Call-related messages are those which direct call control for a call in progress. This chapter overrides portions of section 4.5 of GR-1298-CORE. Encodings for all messages and parameters are provided in Appendix A, “Parameter encodings” or in GR-1299-CORE.

Actions required on the part of the service switching point (SSP) or SCP in generating or responding to these messages are described in GR-1298-CORE. Refer to GR-1298-CORE for full requirements on message processing except where explicitly overridden in the following descriptions.

For each message described in this chapter, a table shows the expected parameters (in order) together with the nature of data to be provided in that parameter. Mandatory parameters are marked in **bold**. Parameters not listed are ignored (passed over) or unsupported (causes this parameter and the remainder of the message to be disregarded). Please refer to Table 2-17 for more information.

4.2 General requirements

The following requirements apply to all messages.

R4-1 **[36]** The population rules for messages in this chapter shall override those in GR-1298-CORE and GR-1299-CORE. Mandatory parameters are marked in bold.

R4-2 **[37]** Extension parameters shall always be optional. Any information transferred in the extension parameter shall be optional; NetworkBuilder shall not require the use of an extension parameter to provide basic trunk-to-trunk call functionality. This allows core functionality to operate in the situation where the SSP communicates with an SCP that cannot support the extension parameters described in Chapter 8, “Extension parameters”.

R4-3 **[38]** SCP call messages are allowed only as an SCP response to an SSP-originated transaction; the SCP may not initiate a transaction to the SSP with a call-related message intended for a particular call in progress or a particular call processing agent.

4.2.1 Routing

The SCP is expected to return to the SSP in the `Analyze_Route` and `Originate_Call` messages a sequence of route choices which may be expressed as trunk groups (`PrimaryTrunkGroup`, `AlternateTrunkGroup`, `SecondAlternateTrunkGroup`), carrier routing information (`Carrier`, `AlternateCarrier`, or `SecondAlternateCarrier`), called party address routing information (`CalledPartyID`, `servingTranslationSystem`, or `univIdx`), or overflow routing information (`GenericAddress OverflowRoutingNumber`). Each of these route choices may provide an STS via a corresponding extension parameter (`servingTranslationSystem`, `PrimaryTrunkGroupSTS`, `AlternateTrunkGroupSTS`, `SecondAlternateTrunkGroupSTS`, or `OverflowRoutingNoSTS`), or through SSP datafill. Outpulse numbers may also be provided, but they are not used to determine the routing for the call.

Trunkgroup or Direct Termination Routing utilizing the `PrimaryTrunkGroup`, `AlternateTrunkGroup`, or `SecondAlternateTrunkGroup` parameters provide for route determination based on NetworkBuilder defined routing tables.

Non-trunkgroup routing (`Carrier` routing or standard routing using a `CalledPartyID`, `servingTranslationSystem`, `univIdx`, or `OverflowRoutingNo`) determines the appropriate STS and `CalledPartyID` for the call (through SSP datafill or returned SCP values) and then uses in-switch translations to determine a route.

In the event that routing information is not returned from the SCP in the `Analyze_Route` message, the SSP shall detect a fatal application error unless the query originated at the Info_Analyzed TDP, in which case the SSP shall attempt to route the call using information it had prior to the query. If routing information is not supplied in the Originate_Call `message`, a `Failure_Outcome` message with a `legid` of 2 is sent to the SCP in a conversation package.

4.2.2 Outpulsing

GR-1298-CORE provides only a single outpulse number which may optionally be used with any of the route choices returned in the `Analyze_Route` or `Originate_Call` messages. UCS DMS-250 feature specifications allow a unique outpulse number be associated with each route choice, requiring up to three outpulse numbers to be returned from the SCP.

The NetworkBuilder implementation thus provides, in addition to the Bellcore `OutpulseNumber` parameter, both an alternate and a second alternate outpulse number. These additional numbers will be encoded as `GenericAddress` parameters. If a route choice is marked to use an outpulse number (other than the `CalledPartyID`), the SSP shall use the outpulse number which matches the route choice (primary to primary, and so forth). If that outpulse number is not included, the original `OutpulseNumber` parameter shall be used. It is a non-fatal application error for route choices to be marked as to require an outpulse number, and none of the outpulse numbers to be included in the `Analyze_Route` message.

4.2.3 Interworking with PSA

Carrier AIN allows interworking with the Programmable Service Architecture (PSA) capability, in that the AIN SCP may direct the DMS-250 to terminate "normal" AIN call processing and enter the PSA environment by querying the PSA Service Control Unit (SCU). To do this, the SCP must return a `Continue` message in response to the original AIN request containing the extension parameter `connectToSCU`. Upon receipt of this message, the DMS-250 shall end AIN call processing and immediately begin PSA call processing for the call; this situation will be identical to the DMS-250 having queried the SCU as a result of trigger table evaluation. Note that PSA interworking is only supported at the Origination_Attempt, Info_Collected, and Info_Analyzed TDPs.

4.3 Analyze_Route message

The `Analyze_Route` message is sent by the SCP in response to a request originated by the SSP originating call model. This message directs the SSP to continue call processing under the direction of the SCP. It is described in section 5.2.1 of GR-1298-CORE and section 5.2.2.2 of GR-1299-CORE.

R4-4 **[39v2]** The SCP shall return the `Analyze_Route` message in a Response package and terminate the TCAP transaction, only if EDPs are not to be armed for the call in progress. Otherwise, the `Analyze_Route` message shall be returned in a Conversation package along with a non-first component containing the `Request_Report_BCM_Event` message for arming EDPs.

R4-5 **[348v3]** Only the `Request_Report_BCM_Event, Furnish_AMA_Information, ACG,` and `Send_Notification` non-call related messages shall be allowed in a non-first component with the `Analyze_Route` message. Please refer to Chapter 5, "Non-call Related Messages" for more information.

R4-6 **[40v5]** The `Analyze_Route` message shall be populated with the parameters shown in Table 4-43.

Table 4-43 Parameters in the Analyze_Route message

Parameter name	Paragraph reference	
	This doc	GR-1299-CORE
ChargeNumber	7.19	6.41
CallingPartyID	7.16	6.36
ChargePartyStationType	7.20	6.42
CalledPartyID	7.15	6.33
OutpulseNumber	7.40	6.98
PrimaryTrunkGroup	7.42	6.104
AlternateTrunkGroup	7.6	6.9
SecondAlternateTrunkGroup	7.45	6.115
Carrier	7.17	6.37
AlternateCarrier	7.5	6.7
SecondAlternateCarrier	7.44	6.114
OverflowBillingIndicator	N/A	6.99
AMAAlternateBillingNumber	7.8	6.10
AMALineNumber	7.9	6.15
AMAslpID	7.11	6.21
AMADigitsDialedWC	7.7	6.14
ExtensionParameter	7.30	6.68
GenericAddressList	7.35	6.80
ForwardCallIndicator	7.33	6.76

R4-7 **[41v2]** The SSP shall populate the call detail record (CDR) with the values returned in the `ChargeNumber`, `CallingPartyID`, and `CalledPartyID` parameters, overwriting any previous values placed there. Note that a value returned in an `AMADigitsDialedWC` parameter for the CDR billing number takes precedence over the value returned in the `ChargeNumber` parameter.

R4-8 **[42v3]** The SSP shall pass to the terminating agent as the calling party identification the value in the `CallingPartyID` parameter, subject to in-switch restrictions on delivery of that identification.

R4-9 **[43]** (Deleted)

R4-10 **[44]** (Deleted).

R4-11 **[253v2]** The `Carrier` parameter shall contain the carrier selection information and the primary carrier identification code (CIC) to which the call is to be routed.

R4-12 **[349]** The `AlternateCarrier` parameter shall contain the carrier selection information and the alternate CIC to which the call is to be routed.

R4-13 **[350]** The `SecondAlternateCarrier` parameter shall contain the carrier selection information and the second alternate CIC to which the call is to be routed.

R4-14 **[857]** The `OverflowBillingIndicator` parameter shall contain a three digit AMA Call Type and three digit Service Feature Identification which, if nonzero, will populate CDR fields AMACALLT and SVCFTR, respectively. A zero value for AMA Call Type will cause the CDR field AMACALLT not to be populated. If an `AMAslpID` parameter is received in the same message, a zero value for Service Feature Identification will cause the CDR field SVCFTR to be populated with a default value of 027; otherwise it will not be populated.

R4-15 **[254]** The `AMADigitsDialedWC` parameter shall appear within an anonymous sequence of this parameter type within the message. Each such parameter shall contain a digit string to be placed within the CDR. The first three digits of the string identify the type of number and thus the field within the CDR to be populated. Up to six of these parameters may be sent in the sequence.

R4-16 **[45]** The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, “Extension parameters” for a description of the currently supported private parameters.

R4-17 **[46v2]** The `GenericAddressList` parameter, if present, may contain any of the following types of addresses:

- alternate outpulse number
- second alternate outpulse number
- overflow routing number
- Dialed Number Inward Service (DNIS)
- Local Number Portability (LNP) ported dialed number

These parameters shall not be transferred directly to the outgoing ISUP IAM; however, the addresses may be sent in the IAM as the called party address.

R4-18 **[47]** (Deleted)

R4-19 **[48]** (Deleted)

R4-20 **[351]** The SCP shall provide replacement Information Digits in the `ChargePartyStationType` parameter.

R4-21 **[49v3]** The SCP shall provide either a route index or a routing number for all queries originated at any TDP other than `Info_Analyzed`. Failure to provide routing information shall be treated as a fatal application error. For queries originated at the `Info_Analyzed` TDP, if no routing information is returned, the SSP shall use the routing information present prior to sending the request.

R4-22 **[50v4]** (Deleted)

R4-23 **[51]** (Deleted)

R4-24 **[352]** The SCP shall indicate that a Local Number Portability (LNP) check has been performed at the SCP by setting bit M of the `ForwardCallIndicator` parameter.

R4-25 **[851]** (Deleted)

R4-26 **[852]** (Deleted)

4.4 Cancel_Resource_Event message

The `Cancel_Resource_Event` message is sent by the SCP when the SSP is processing an outstanding `Send_To_Resource` operation initiated by the SCP. This message directs the SSP to discontinue caller interaction and report to the SCP for further instructions. It is described in section 5.2.2.5 of GR-1298-CORE and section 5.5 of GR-1299-CORE.

R4-27 **[255]** The `Cancel_Resource_Event` message shall be sent in a Conversation package. The SSP shall maintain the transaction upon receipt of this message. The SSP shall respond with a `Resource_Clear` message unless a `Resource_Clear` has been sent already to indicate the end of the caller interaction, in which case the `Cancel_Resource_Event` message shall be ignored.

R4-28 **[256]** The `Cancel_Resource_Event` message shall be populated with the parameters shown in Table 4-44.

Table 4-44 Parameters in the Cancel_Resource_Event message

Parameter name	Paragraph reference	
	This doc	GR-1299-CORE
ExtensionParameter	7.30	6.68

R4-29 **[257]** The SCP shall not include any other message with the `Cancel_Resource_Event` message.

R4-30 **[258]** The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, "Extension parameters" for a description of the currently supported private parameters.

4.5 Continue message

The `Continue` message is sent by the SCP in response to a request originated by the SSP originating call model. This message directs the SSP to continue in-switch call processing as if the call had not triggered a query to the SCP. It is described in section 5.2.6 of GR-1298-CORE and section 5.2.2.8 of GR-1299-CORE.

R4-31 **[52]** The SCP shall return the `Continue` message in a Response package and terminate the TCAP transaction, only if EDPs are not to be armed for the call in progress. Otherwise, the `Continue` message shall be returned in a Conversation package along with a non-first component containing the `Request_Report_BCM_Event` message for arming EDPs.

R4-32 **[353v2]** Only the `Request_Report_BCM_Event, ACG, a`nd `Send_Notification` non-call related messages shall be allowed in a non-first component with the `Continue` message. Please refer to Chapter 5, "Non-call Related Messages" for more information.

R4-33 **[53v4]** The `Continue` message shall be populated with the parameters shown in Table 4-45.

Table 4-45 Parameters in the Continue message

Parameter name	Paragraph reference	
	This doc	GR-1299-CORE
AMAslpID	7.11	6.21
AMADigitsDialedWC	7.7	6.14
ExtensionParameter	7.30	6.68

R4-34 **[54]** (Deleted)

R4-35 **[55]** (Deleted)

R4-36 **[56v2]** The SSP shall continue TDP processing at the point in call where it left off. Immediately after receiving a `Continue` message, the SSP may trigger the same call again at the same TDP.

R4-37 **[259]** The `AMADigitsDialedWC` parameter shall appear within an anonymous sequence of this parameter type within the message. Each such parameter shall contain a digit string to be placed within the CDR. The first three digits of the string identify the type of number and thus the field within the CDR to be populated. Up to six of these parameters may be sent in the sequence.

R4-38 **[57]** The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, “Extension parameters” for a description of the currently supported private parameters.

4.6 Disconnect message

The `Disconnect` message is sent by the SCP in response to a request originated by the SSP. It directs the SSP to discontinue the call. It is described in section 5.2.7 of GR-1298-CORE and section 5.2.2.10 of GR-1299-CORE.

R4-39 **[58]** The `Disconnect` message shall be sent in a Response package. The SSP shall close the transaction upon receipt of this message.

R4-40 **[526v2]** Only the `ACG` and `Send_Notification` non-call related messages shall be allowed in a non-first component with the `Disconnect` message. Please refer to Chapter 5, “Non-call Related Messages” for more information.

Nortel Networks Proprietary and Confidential

R4-41 **[59v4]** The `Disconnect` message shall be populated with the parameters shown in Table 4-46.

Table 4-46 Parameters in the Disconnect message

Parameter name	Paragraph reference	
	This doc	GR-1299-CORE
AMAslpID	7.11	6.21
AMADigitsDialedWC	7.7	6.14
ExtensionParameter	7.30	6.68

R4-42 **[60]** (Deleted)

R4-43 **[61v2]** Upon receipt of the `Disconnect` message, the SSP shall discontinue call processing on the incoming agent. The SSP may treat the call depending on other feature requirements.

R4-44 **[260]** The `AMADigitsDialedWC` parameter shall appear within an anonymous sequence of this parameter type within the message. Each such parameter shall contain a digit string to be placed within the CDR. The first three digits of the string identify the type of number and thus the field within the CDR to be populated. Up to six of these parameters may be sent in the sequence.

R4-45 **[62]** The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, “Extension parameters” for a description of the currently supported private parameters.

4.7 Send_To_Resource message

The `Send_To_Resource` message is sent by the SCP in response to a request from the SSP. It is described in section 5.5 of GR-1298-CORE and section 5.2.2.18 of GR-1299-CORE.

R4-46 **[63v2]** The `Send_To_Resource` message shall be transmitted in a Conversation package unless the `DisconnectFlag` parameter is present, in which case is shall be transmitted in a Response package.

R4-47 **[64v2]** A `Send_To_Resource` message received by the SSP in a Response package with no `DisconnectFlag` parameter shall be treated as a non-fatal application error. The SSP shall infer the presence of the `DisconnectFlag` parameter in this case. If the message is received in a Conversation package with the `DisconnectFlag` parameter present, it shall be treated as a fatal application error.

R4-48 **[261]** A `Send_To_Resource` message received by the SSP in a Response package with the `DestinationAddress` parameter present shall be treated as a fatal application error.

R4-49 **[527v3]** Only the `ACG`, `Furnish_AMA_Information`, and `Send_Notification` non-call related messages shall be allowed in a non-first component with the `Send_To_Resource` message. Please refer to Chapter 5, “Non-call Related Messages” for more information.

R4-50 **[65v5]** The `Send_To_Resource` message shall be populated with the parameters shown in Table 4-47.

Table 4-47 Parameters in the Send_To_Resource message

Parameter name	Paragraph reference	
	This doc	GR-1299-CORE
ResourceType	7.43	6.108
STRParameterBlock	7.47	6.124
DisconnectFlag	N/A	6.57
AnswerIndicator	N/A	6.24
PrimaryBillingIndicator	N/A	6.103
AMAslpID	7.11	6.21
AMADigitsDialedWC	7.7	6.14
DestinationAddress	7.26	6.55
AMAMeasure	7.9	6.16
ExtensionParameter	7.30	6.68

R4-51 **[66v3]** The `ResourceType` parameter shall be encoded as “play announcement” or “play announcement and collect digits” or “flex parameter block” when no `DestinationAddress` parameter is present. The use of other values shall cause a non-fatal application error. If the `DestinationAddress` parameter is present, the `ResourceType` is unmodified and passed to the IP.

R4-52 **[67v2]** If the `ResourceType` is “play announcement”, the `STRParameterBlock` shall take the form of an `AnnouncementBlock`, and shall contain only an `UninterAnnounceBlock`.

R4-53 **[262]** If the `ResourceType` is “play announcement and collect digits”, the `STRParameterBlock` shall take the form of an `AnnouncementDigitBlock`.

R4-54 **[263]** If the `ResourceType` is “play announcement and collect digits”, the SCP shall indicate in the `AnnouncementDigitBlock` a tone or announcement to be used to prompt the subscriber. The SSP shall treat as a fatal application error any request to collect digits not prefaced by a prompt tone or announcement.

R4-55 **[264v2]** If the `ResourceType` is “play announcement and collect digits”, the SCP shall indicate in the `AnnouncementDigitBlock` either a fixed, variable or “normal” number of digits to collect.

R4-56 **[265v2]** Within an `AnnouncementDigitBlock`, the SCP may indicate that the SSP is to collect “up to” a maximum number of digits, as a variation of the standard “variable number of digits”, where the maximum number of digits is fixed by the size of the `CollectedDigits` parameter. In this case, the SSP shall collect from 0 up to the indicated number of digits as a variable-length digit string, and report that string of digits to the SCP. The `AnnouncementDigitBlock` is described in sections 7.47 and A.2.7.

R4-57 **[661]** A request by the SCP for a “normal” number of address digits to be collected requires that the SSP collects the appropriate number of address digits based on the dialling plan in force at the time of the collection of the address. (GR-1298-CORE [469]).

R4-58 **[68v2]** Only the first three `AnnounceElements` of the `UninterAnnounceBlock` or `InterAnnounceBlock` shall be used by the SSP; remaining elements shall be ignored. In addition, any info digits, if present, shall be ignored.

R4-59 **[69v3]** If the `AnswerIndicator` parameter is present, the SSP shall provide answer supervision to the originating agent while the caller is connected to the announcement or IP.

R4-60 **[858]** The `PrimaryBillingIndicator` parameter shall contain a three digit AMA Call Type and three digit Service Feature Identification which, if nonzero, will populate CDR fields AMACALLT and SVCFTR, respectively. A zero value for AMA Call Type will cause the CDR field AMACALLT to not be populated. If an `AMAslpID` parameter was received in the same message, a zero value for Service Feature Identification will cause the CDR field SVCFTR to be populated with a default value of 027, otherwise it will not be populated.

R4-61 **[354]** (Deleted)

R4-62 **[528]** The `AMADigitsDialedWC` parameter shall appear within an anonymous sequence of this parameter type within the message. Each

such parameter shall contain a digit string to be placed within the CDR. The first three digits of the string identify the type of number and thus the field within the CDR to be populated. Up to six of these parameters may be sent in the sequence.

R4-63 **[529]** The AMAMeasure parameter shall indicate that a duration time measurement is required as specified by the value of the parameter only if the value is connectTimeRecordedDestinationSSP. All other values of the parameter are ignored.

R4-64 **[530]** The AMAMeasure parameter may alter the value of the answer type field of the CDR for an 1129-style STR connection.

R4-65 **[70]** The ExtensionParameter, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, "Extension parameters" for a description of the currently supported private parameters.

4.7.1 Intelligent Peripheral Requirements

R4-66 **[266v4]** If the DestinationAddress parameter is present, the SSP shall route the call to that address rather than sending the call to an in-switch resource. Note that the terminating agent for an IP connection must be an ISUP IMT or PRI with the correct provisioning.

R4-67 **[531]** The DestinationAddress parameter shall not be processed in response to a Termination_Attempt request message.

R4-68 **[532]** If the ResourceType is "flex parameter block" and no DestinationAddress is present, the STRParameter shall be encoded as a FlexParameterBlock using the NTFlexParameterBlock encoding as specified in section A.6. All other encodings result in a non-fatal application error.

R4-69 **[533]** The NTFlexParameterBlock shall be encoded using the VIPParameterBlock encoding as specified in A.6. All other encodings result in a non-fatal application error.

4.8 Authorize_Termination

The Authorize_Termination message is sent by the SCP in response to a request originated by the SSP terminating call model. This message directs the SSP to allow the indicated termination to progress through the TCM. It is described in section 5.2.4 of GR-1298-CORE and section 5.2.2.3 of GR-1299-CORE.

R4-70 **[534]** The SCP shall return the Authorize_Termination in a Response package. The SSP shall close the transaction upon receipt of this message.

R4-71 **[535v2]** Only the ACG and Send_Notification non-call related messages shall be allowed in a non-first component with the

`Authorize_Termination` message. Please refer to Chapter 5, “Non-call Related Messages” for more information.

R4-72 **[536v3]** The `Authorize_Termination` message shall be populated with the parameters shown in Table 4-48.

Table 4-48 Parameters in the Authorize_Termination message

Parameter name	Paragraph reference	
	This doc	GR-1299-CORE
CallingPartyID	7.16	6.36
DisplayText	7.27	6.58
AMAslpID	7.11	6.21
AMADigitsDialedWC	7.7	6.14
ExtensionParameter	7.30	6.68

R4-73 **[537]** The SSP shall populate the call detail record (CDR) with the values returned in the `CallingPartyID` parameter, overwriting any previous values placed there.

R4-74 **[538]** The SSP shall pass to the terminating agent as the calling party identification the value in the `CallingPartyID` parameter, subject to in-switch restrictions on delivery of that identification. The SCP shall set the Presentation Indicator field of this parameter to “presentation restricted” in order to restrict delivery of the calling party’s ID. If the `CallingPartyID` parameter is not present, the SSP shall use in-switch logic to determine whether to deliver the identification.

R4-75 **[539]** The `AMADigitsDialedWC` parameter shall appear within an anonymous sequence of this parameter type within the message. Each such parameter shall contain a digit string to be placed within the CDR. The first three digits of the string identify the type of number and thus the field within the CDR to be populated. Up to six of these parameters may be sent in the sequence.

R4-76 **[540]** The `DisplayText` parameter shall provide display data that is to be sent across the network to the end user. This is only used with PRI terminations.

R4-77 **[541]** The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, “Extension parameters” for a description of the currently supported private parameters.

4.9 Connect_To_Resource

The Connect_To_Resource message is sent by the SCP in response to a request from the SSP. It is described in section 21.6.2 of GR-1298-CORE and section 5.2.2.7 of GR-1299-CORE.

R4-78 **[542]** The Connect_To_Resource message shall be transmitted in a Conversation package unless the DisconnectFlag parameter is present, in which case is shall be transmitted in a Response package.

R4-79 **[543]** A Connect_To_Resource message received by the SSP in a Response package with no DisconnectFlag parameter shall be treated as a non-fatal application error. The SSP shall infer the presence of the DisconnectFlag parameter in this case. If the message is received in a Conversation package with the DisconnectFlag parameter present, it shall be treated as a fatal application error.

R4-80 **[544]** A Connect_To_Resource message received by the SSP in a Response package with the DestinationAddress parameter present shall be treated as a fatal application error.

R4-81 **[545]** Only the ACG and Send_Notification non-call related messages shall be allowed in a non-first component with the Connect_To_Resource message. Please refer to Chapter 5, "Non-call Related Messages" for more information.

R4-82 **[546v2]** The Connect_To_Resource message shall be populated with the parameters shown in Table 4-47.

Table 4-49 Parameters in the Connect_To_Resource message

Parameter name	Paragraph reference	
	This doc	GR-1299-CORE
ResourceType	7.43	6.108
STRParameterBlock	7.47	6.124
LegID	7.39	6.90
DisconnectFlag	N/A	6.57
AMAslpID	7.11	6.21
AMADigitsDialedWC	7.7	6.14
DestinationAddress	7.26	6.55
AMAMeasure	7.9	6.16
ExtensionParameter	7.30	6.68

R4-83 **[547]** The `ResourceType` parameter shall be encoded as “play announcement” or “play announcement and collect digits” or “flex parameter block” when no `DestinationAddress` parameter is present. The use of other values shall cause a non-fatal application error. If the `DestinationAddress` parameter is present, the `ResourceType` is unmodified and passed to the IP.

R4-84 **[548]** It the `ResourceType` is “play announcement”, the `STRParameterBlock` shall take the form of an `AnnouncementBlock`, and shall contain only an `UninterAnnounceBlock`.

R4-85 **[549]** If the `ResourceType` is “play announcement and collect digits”, the `STRParameterBlock` shall take the form of an `AnnouncementDigitBlock`.

R4-86 **[550]** If the `ResourceType` is “play announcement and collect digits”, the SCP shall indicate in the `AnnouncementDigitBlock` a tone or announcement to be used to prompt the subscriber. The SSP shall treat as a fatal application error any request to collect digits not prefaced by a prompt tone or announcement.

R4-87 **[551v2]** If the `ResourceType` is “play announcement and collect digits”, the SCP shall indicate in the `AnnouncementDigitBlock` either a fixed, variable or “normal” number of digits to collect.

R4-88 **[552]** Within an `AnnouncementDigitBlock`, the SCP may indicate that the SSP is to collect “up to” a maximum number of digits, as a variation of the standard “variable number of digits”, where the maximum number of digits is fixed by the size of the `CollectedDigits` parameter. In this case, the SSP shall collect from 0 up to the indicated number of digits as a variable-length digit string, and report that string of digits to the SCP. Changes to the `AnnouncementDigitBlock` are described in sections 7.47 and A.2.7.

R4-89 **[662]** A request by the SCP for a “normal” number of address digits to be collected requires that the SSP collects the appropriate number of address digits based on the dialling plan in force at the time of the collection of the address. (GR-1298-CORE [469]).

R4-90 **[553]** Only the first three `AnnounceElements` of the `UninterAnnounceBlock` or `InterAnnounceBlock` shall be used by the SSP; remaining elements shall be ignored. In addition, any info digits, if present, shall be ignored.

R4-91 **[554]** The `LegID` parameter indicates which leg of a call segment interacts with the indicated resource.

R4-92 **[555]** The `AMADigitsDialedWC` parameter shall appear within an anonymous sequence of this parameter type within the message. Each such parameter shall contain a digit string to be placed within the CDR.

The first three digits of the string identify the type of number and thus the field within the CDR to be populated. Up to six of these parameters may be sent in the sequence.

R4-93 **[556]** The `AMAMeasure` parameter shall indicate that a duration time measurement is required as specified by the value of the parameter only if the value is `connectTimeRecordedDestinationSSP`. All other values of the parameter are ignored.

R4-94 **[557]** The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, “Extension parameters” for a description of the currently supported private parameters.

4.9.1 Intelligent Peripheral Requirements

R4-95 **[558]** If the `ResourceType` is “flex parameter block” and no `DestinationAddress` is present, the `STRParameter` shall be encoded as a `FlexParameterBlock` using the `NTFlexParameterBlock` encoding as specified in section A.6. All other encodings result in a non-fatal error.

R4-96 **[559]** The `NTFlexParameterBlock` shall be encoded using the `VIPParameterBlock` encoding as specified in A.6. All other encodings result in a non-fatal error.

R4-97 **[560v2]** If the `DestinationAddress` parameter is present, the SSP shall route the call to that address rather than sending the call to an in-switch resource. Note that the terminating agent for an IP connection must be an ISUP IMT or PRI with the correct provisioning.

4.10 Collect_Information

The `Collect_Information` message is sent by the SCP in response to a request message originating at the SSP. It is described in section 5.2.3 of GR-1298-CORE and section 5.2.2.6 of GR-1298-CORE.

R4-98 **[561]** The SCP shall return the `Collect_Information` message in a Response package and terminate the TCAP transaction, only if EDPs are not to be armed for the call in progress. Otherwise, the `Collect_Information` message shall be returned in a Conversation package along with a non-first component containing the `Request_Report_BCM_Event` message for arming EDPs.

R4-99 **[562]** Only the `Request_Report_BCM_Event, ACG,` and `Send_Notification` non-call related messages shall be allowed in a non-first component with the `Collect_Information` message. Please refer to Chapter 5, “Non-call Related Messages” for more information.

R4-100 [563v2] The `Collect_Information` message shall be populated with the parameters shown in Table 4-50.

Table 4-50 Parameters in the Collect_Information message

Parameter name	Paragraph reference	
	This doc	GR-1299-CORE
CallingPartyID	7.16	6.36
AMAslpID	7.11	6.21
AMADigitsDialedWC	7.7	6.14
ExtensionParameter	7.30	6.68

R4-101 [564] The SSP shall populate the call detail record (CDR) with the values returned in the `CallingPartyID` parameter, overwriting any previous values placed there.

R4-102 [565] The SSP shall pass to the terminating agent as the calling party identification the value in the `CallingPartyID` parameter, subject to in-switch restrictions on delivery of that identification. The SCP shall set the Presentation Indicator field of this parameter to "presentation restricted" in order to restrict delivery of the calling party's ID. If the `CallingPartyID` parameter is not present, the SSP shall use in-switch logic to determine whether to deliver the identification.

R4-103 [566] The `AMADigitsDialedWC` parameter shall appear within an anonymous sequence of this parameter type within the message. Each such parameter shall contain a digit string to be placed within the CDR. The first three digits of the string identify the type of number and thus the field within the CDR to be populated. Up to six of these parameters may be sent in the sequence.

R4-104 [567] The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, "Extension parameters" for a description of the currently supported private parameters.

4.11 Call_Info_To_Resource

The `Call_Info_To_Resource` message is used to provide a response to a `Call_Info_From_Resource`. This only occurs during an active STR-Connection. It is described in section 5.2.2.4 of GR-1299-CORE and section 4 of GR-1129-CORE.

R4-105 [568] The `Call_Info_To_Resource` message shall be sent in a Return result component.

R4-106 [569] The `Call_Info_To_Resource` message shall be populated with the parameters shown in Table 4-51.

Table 4-51 Parameters in the Call_Info_To_Resource message

Parameter name	Paragraph reference	
	This doc	GR-1299-CORE
ResourceType	7.43	6.108
StrParameterBlock	7.47	6.124
ExtensionParameter	7.30	6.68

R4-107 [570] The `ResourceType` and/or `StrParameterBlock` parameters are unmodified and passed to the IP.

R4-108 [571] The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, “Extension parameters” for a description of the currently supported private parameters.

R4-109 [853] The SCP shall not include any other message with the `Call_Info_To_Resource` message.

4.12 Originate_Call message

The `Originate_Call` message is sent by the SCP in response to a request originated by the SSP originating call model. This message directs the SSP to continue call processing under the direction of the SCP. It is described in section 21.6.1 of GR-1298-CORE and section 5.2.2.16 of GR-1299-CORE.

R4-110 [787] The `Originate_Call` message shall be returned in a Conversation package along with a non-first component containing the `Request_Report_BCM_Event` message for arming EDPs (GR-1298-CORE R21-268 [2274]).

R4-111 [854] Only the `Request_Report_BCM_Event`, `ACG` and `Send_Notification` non-call related messages shall be allowed in a non-first component with the `Originate_Call` message. Please refer to Chapter 5, “Non-call Related Messages” for more information.

R4-112 [788] The `Originate_Call` message shall be populated with the parameters shown in Table 4-52.

Table 4-52 Parameters in the Originate_Call message

Parameter name	Paragraph reference	
	This doc	GR-1299-CORE
ChargeNumber	7.19	6.41
CallingPartyID	7.16	6.36
ChargePartyStationType	7.20	6.42
CalledPartyID	7.15	6.33
OutpulseNumber	7.40	6.98
PrimaryTrunkGroup	7.42	6.104
AlternateTrunkGroup	7.6	6.9
SecondAlternateTrunkGroup	7.45	6.115
Carrier	7.17	6.37
AlternateCarrier	7.5	6.7
SecondAlternateCarrier	7.44	6.114
AMAslpID	7.11	6.21
AMADigitsDialedWC	7.7	6.14
ExtensionParameter	7.30	6.68
GenericAddressList	7.35	6.80
ForwardCallIndicator	7.33	6.76
DisplayText	7.24	6.58
AMAAlternateBillingNumber	7.8	6.10
AMALineNumber	7.9	6.15

R4-113 [789] The SSP shall populate the call detail record (CDR) with the values returned in the `ChargeNumber`, `CallingPartyID`, and `CalledPartyID` parameters. Note that a value returned in an `AMADigitsDialedWC` parameter for the CDR billing number takes precedence over the value returned in the `ChargeNumber` parameter.

R4-114 [790] The SSP shall pass to the terminating agent as the calling party identification the value in the `CallingPartyID` parameter, subject to in-switch restrictions on delivery of that identification. The SCP

shall set the Presentation Indicator field of this parameter to "presentation restricted" in order to restrict delivery of the calling party's ID. If the `CallingPartyID` parameter is not present, the SSP shall use in-switch logic to determine whether to deliver the identification.

R4-115 [791] The `OutpulseNumber` parameter shall only be included if the Number to Outpulse flag of at least one of the trunk group parameters (`PrimaryTrunkGroup`, and so forth) is set. If no trunk group parameters are transmitted, this parameter shall not be transmitted.

R4-116 [792] The `PrimaryTrunkGroup`, `AlternateTrunkGroup`, and `SecondAlternateTrunkGroup` parameters shall be filled as per section 7.2.2.

R4-117 [793] The `Carrier` parameter shall contain the carrier selection information and the primary carrier identification code (CIC) to which the call is to be routed.

R4-118 [794] The `AlternateCarrier` parameter shall contain the carrier selection information and the alternate CIC to which the call is to be routed.

R4-119 [795] The `SecondAlternateCarrier` parameter shall contain the carrier selection information and the second alternate CIC to which the call is to be routed.

R4-120 [796] The `AMADigitsDialedWC` parameter shall appear within an anonymous sequence of this parameter type within the message. Each such parameter shall contain a digit string to be placed within the CDR. The first three digits of the string identify the type of number and thus the field within the CDR to be populated. Up to six of these parameters may be sent in the sequence.

R4-121 [797] The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, "Extension parameters" for a description of the currently supported private parameters.

R4-122 [798] The `GenericAddressList` parameter, if present, may contain any of the following types of addresses:

- alternate outpulse number
- second alternate outpulse number
- overflow routing number
- Dialed Number Inward Service (DNIS)
- Local Number Portability (LNP) ported dialed number

These parameters shall not be transferred directly to the outgoing ISUP IAM; however, the addresses may be sent in the IAM as the called party address.

Nortel Networks Proprietary and Confidential

R4-123 [799] The SSP shall outpulse the `CalledPartyID`, unless one or more of the outpulse number parameters is present and the Number to Outpulse field of the trunk group parameter (for the route currently selected) is set, in which case the specified outpulse number shall be outpulsed.

R4-124 [800] The SCP may supply information digits in the `ChargePartyStationType` parameter.

R4-125 [801] The SCP shall always provide either a trunk group and outpulse number or a routing number. If no routing information is returned, a `Failure_Outcome` message with a `LegID` parameter of 2 will be sent to the SCP in a Conversation package.

R4-126 [802] The SCP may return from one to three route choices in the trunk group parameters. If none are present, the SSP shall use the called party address (or the address contained in the `CalledPartyID` parameter, if present) along with a translations scheme derived from each of the carrier parameters to determine the routing. If no carrier parameters are present, the SSP shall use the called party address (or the address contained in the `CalledPartyID` parameter, if present) to determine the routing using an `STS` or `univIdx` sent in an extension parameter, or in-switch default `servTranslationScheme` or `univIdx`.

4.13 Merge_Call message

The `Merge_Call` message is sent by the SCP in response to a request originated by the SSP originating call model. This message directs the SSP to merge all existing legs (either two or three) into a single call. It is described in section 21.6.6 of GR-1298-CORE and section 5.2.2.13 of GR-1299-CORE.

R4-127 [803] If a `Merge_Call` message is received in any CC other than CC4 or CC6, the SSP shall treat it as a fatal application error.

R4-128 [853] Only the `Request_Report_BCM_Event`, `ACG` and `Send_Notification` non-call related messages shall be allowed in a non-first component with the `Merge_Call` message. Please refer to Chapter 5, “Non-call Related Messages” for more information.

R4-129 [804] The `Merge_Call` message shall not be populated with any parameters (including extension parameters).

4.14 Disconnect_Leg message

The `Disconnect_Leg` message is sent by the SCP in response to a request originated by the SSP originating call model. This message directs the SSP to disconnect the specified leg from the call. It is described in section 21.6.7 of GR-1298-CORE and section 5.2.2.11 of GR-1299-CORE.

R4-130 [805] The `Disconnect_Leg` message shall only be accepted from CC4 through CC10, inclusive. A fatal application error is detected for any other CC.

R4-131 [852] Only the `Request_Report_BCM_Event`, `ACG` and `Send_Notification` non-call related messages shall be allowed in a non-first component with the `Disconnect_Leg` message. Please refer to Chapter 5, "Non-call Related Messages" for more information.

R4-132 [806] The `Disconnect_Leg` message shall be populated with the parameters shown in Table 4-53.

Table 4-53 Parameters in the Disconnect_Leg message

Parameter name	Paragraph reference	
	This doc	GR-1299-CORE
LegID	7.39	6.90
CsID	7.25	6.53
ExtensionParameter	7.30	6.68

R4-133 [807] The `LegID` parameter shall be populated with either 0, 1, or 2, depending on the which leg is to be disconnected.

R4-134 [808] The `CsID` parameter shall be populated with the call segment to which the leg to be disconnected is attached (either 1 or 2).

R4-135 [809] The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, "Extension parameters" for a description of the currently supported private parameters.

4.15 Acknowledge message

The `Acknowledge` message is sent by the SCP in response to a request originated by the SSP originating call model. This message allows the SCP to inform the SSP that it has received the request but that it does not want the SSP to perform any state changing actions. It is described in section 21.6.6 of GR-1298-CORE and section 5.2.2.13 of GR-1299-CORE.

R4-136 [850] The `Acknowledge` message shall be returned in a Conversation package along with a non-first component containing the `Request_Report_BCM_Event` message for arming EDPs (GR-1298-CORE R21-268 [2274]).

R4-137 [849] If an `Acknowledge` message is received in any CC other than CC4 or CC6, the SSP shall treat it as a fatal application error.

R4-138 [851] Only the `Request_Report_BCM_Event`, `ACG` and `Send_Notification` non-call related messages shall be allowed in a non-first component with the `Acknowledge` message. Please refer to Chapter 5, “Non-call Related Messages” for more information.

R4-139 [810] The `Acknowledge` message shall be populated with the parameters shown in Table 4-54.

Table 4-54 Parameters in the Acknowledge message

Parameter name	Paragraph reference	
	This doc	GR-1299-CORE
CsID	7.25	6.53
ExtensionParameter	7.30	6.68

R4-140 [811] The `CsID` parameter shall be populated with the call segment on which the event occurred (either 1 or 2).

R4-141 [812] The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, “Extension parameters” for a description of the currently supported private parameters.

Chapter 5: Non-call Related Messages

5.1 Introduction

This section provides an overview of the non-call related messages supported by the NetworkBuilder implementation of Carrier AIN, together with rules for populating the messages with parameters. Non-call related messages are those messages not directly involved in call control. This chapter overrides portions of GR-1298-CORE. Encodings for all messages and parameters are provided in Appendix A or in GR-1299-CORE.

Actions required on the part of the service switching point (SSP) or SCP in generating or responding to these messages are described in GR-1298-CORE. Refer to GR-1298-CORE for full requirements on message processing except where explicitly overridden in the following descriptions.

For each message described in this chapter, a table shows the expected parameters (in order) together with the nature of data to be provided in that parameter. Mandatory parameters are marked in **bold**. Parameters not listed are ignored (passed over) or unsupported (causes this parameter and the remainder of the message to be disregarded). Please refer to Table 2-17 for more information.

5.2 General requirements

The following requirements apply to all messages.

R5-1 **[355v3]** The SCP shall send up to four non-call related messages in any TCAP package, including SCP response messages.

R5-2 **[356v4]** Except for the `ACG_Global_Control_Restore` and `ACG` messages, the SSP only supports receiving non-call related messages in the non-first components of a TCAP package. The first component of a TCAP response or conversation package shall contain an SCP call-related response message to an SSP-originated transaction.

5.3 SCP non-call related messages

5.3.1 Request_Report_BCM_Event message

The `Request_Report_BCM_Event` message is sent by the SCP to indicate to the SSP which event detection points (EDPs) should be armed to send an EDP-Request or EDP-Notification to the SCP. This message is described in section 5.3 of GR-1298-CORE and section 5.3.16 of GR-1299-CORE.

R5-3 **[357v2]** The SCP shall return the `Request_Report_BCM_Event` message as any non-first component in a Conversation package.

R5-4 **[358]** The `Request_Report_BCM_Event` message shall be populated with the parameters shown in Table 5-55.

Table 5-55 Parameters in the Request_Report_BCM_Event message

Parameter name	Paragraph reference	
	This doc	GR-1299-CORE
EDPRequest	7.29	6.63
EDPNotification	7.28	6.62
ONoAnswerTimer	N/A	6.95
ExtensionParameter	7.30	6.68
TimeoutTimer	N/A	6.129

R5-5 **[359]** The `EDPRequest` parameter shall contain the list of EDPs that should be armed to send an EDP-Request.

R5-6 **[360]** The `EDPNotification` parameter shall contain the list of EDPs that should be armed to send an EDP-Notification.

R5-7 **[361]** The `ONoAnswerTimer` parameter shall be filled with the value, in seconds, of the SSP originating no answer timer. The SSP shall use this value to determine when the `O_No_Answer` EDP has been encountered.

R5-8 **[362]** Deleted.

R5-9 **[363]** The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, “Extension parameters” for a description of the currently supported private parameters.

R5-10 **[364v2]** The value of the `TimeoutTimer` shall be filled with the value, in minutes, of the SSP terminating timeout timer.

5.3.2 Send_Notification message

The Send_Notification message is sent by the SCP to indicate to the SSP that the SCP is to be notified of the termination of the call. This message is described in section 5.8 of GR-1298-CORE and section 5.3.17 of GR-1299-CORE.

R5-11 **[572]** Deleted.

R5-12 **[663]** The SCP shall return the Send_Notification message as any non-first component in a Conversation or Response package.

R5-13 **[664]** If the SSP receives multiple Send_Notification requests for the same call and each request contains an unique TCAP transaction ID, the SSP will send one Termination_Notification message for each request with a unique transaction ID when an event occurs that releases the call.

R5-14 **[665]** If the SSP receives multiple Send_Notification requests within the same TCAP transaction, the SSP will only reply to the last Send_Notification received in the transaction.

R5-15 **[666]** The number of Send_Notification messages the SSP can receive is controlled by inswitch datafill.

R5-16 **[667]** The Send_Notification message shall be populated with the parameters shown in Table 5-56.

Table 5-56 Parameters in the Send_Notification message

Parameter name	Paragraph reference	
	This doc	GR-1299-CORE
EchoData	N/A	6.60
ExtensionParameter	7.30	6.68

R5-17 **[668]** The EchoData parameter is used to associate the SCP Send_Notification message with the SSP Termination_Notification message.

R5-18 **[669]** The ExtensionParameter, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, “Extension parameters” for a description of the currently supported private parameters.

5.3.3 ACG message

The ACG message is sent by the SCP to indicate to the SSP that automatic code gapping shall be initiated. This message is described in section 14 of GR-1298-CORE and section 5.31 of GR-1299-CORE.

R5-19 **[573]** Deleted.

R5-20 **[670]** The ACG message can be received as any non-first component in a Conversation or Response package.

R5-21 **[671]** The ACG message can be received in a unidirectional non-call related package.

R5-22 **[672]** The ACG message is received in an invoke-last component including an operation code.

R5-23 **[673]** The ACG message is not correlated to any other message.

R5-24 **[674]** The ACG message shall be populated with the parameters shown in Table 5-57.

Table 5-57 Parameters in the ACG message

Parameter name	Paragraph reference	
	This doc	GR-1299-CORE
ControlCauseIndicator	N/A	6.50
GapDuration	N/A	6.77
GapInterval	N/A	6.78
TranslationType	N/A	6.131
GlobalTitleAddress	N/A	6.84
ExtensionParameter	7.30	6.68

R5-25 **[675]** The ControlCauseIndicator parameter indicates whether the control is an SCP overload control or an SMS initiated control.

R5-26 **[676]** The ControlCauseIndicator parameter contains the number of digits to which the control is applied.

R5-27 **[677]** The GapDuration parameter contains the length in seconds that an ACG control should be applied before it times out and is removed from the control list by the SSP. Valid gap durations are discussed in GR-1299-CORE, section 6.77.

R5-28 **[678]** The GapInterval parameter contains the minimum length in seconds that the SCP must wait before sending another query of the type under the control. Valid gap intervals for SCP Overload Controls are discussed in GR-1299-CORE, section 6.78.

R5-29 **[679]** The TranslationType parameter is an integer from 0 to 255 indicating the translation type of the ACG control.

R5-30 **[680]** The `GlobalTitleAddress` parameter is a string of digits indicating the global title address of the ACG control.

R5-31 **[681]** The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, "Extension parameters" for a description of the currently supported private parameters.

5.3.4 ACG_Global_Ctrl_Restore message

The `ACG_Global_Ctrl_Restore` message is used to control removal of a large selection of controls all at once. (The SSP has the ability to remove the controls one by one or a large selection all at one time from the ACG control lists.) This message is described in section 14 of GR-1298-CORE and section 5.3.2 of GR-1299-CORE.

R5-32 **[682]** The SCP shall send the `ACG_Global_Ctrl_Restore` message as a non-call related query message in an invoke-last component.

R5-33 **[683]** The SSP shall remove all controls according to this message, regardless of the SCP that initiated the control (including an LNP SCP).

R5-34 **[684]** The `ACG_Global_Ctrl_Restore` message shall be populated with the parameters shown in Table 5-58.

Table 5-58 Parameters in the ACG_Global_Ctrl_Restore message

Parameter name	Paragraph reference	
	This doc	GR-1299-CORE
ACGGlobalOverride	7.4	6.3
ExtensionParameter	7.30	6.68

R5-35 **[685]** The `ACGGlobalOverride` parameter re-initializes the ACG control list according the option provided. Valid ACG Global Override options are discussed in section 7.4, "ACGGlobalOverride parameter".

R5-36 **[686]** The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, "Extension parameters" for a description of the currently supported private parameters.

5.3.5 Furnish_AMA_Information message

The `Furnish_AMA_Information` message is used by the SCP to deliver information to the SSP, requesting to have this information appended to the

call record. This message is described in section 9.2.1 of GR-1298-CORE and section 5.3.8 of GR-1299-CORE.

R5-37 **[859]** The SCP shall return the `Furnish_AMA_Information` message as any non-first component in a Conversation or Response package.

R5-38 **[860]** The call-related message of the TCAP package in which `Furnish_AMA_Information` is included must be `Analyze_Route` or `Send_To_Resource`.

R5-39 **[861]** The `Furnish_AMA_Information` message shall be populated with the parameters shown in Table 5-59.

Table 5-59 Parameters in the Furnish_AMA_Information message

Parameter name	Paragraph reference	
	This doc	GR-1299-CORE
AMABAFModules	N/A	6.11
AMASetHexABIndicator	N/A	6.19
ExtensionParameter	7.30	6.68

R5-40 **[862]** The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, “Extension parameters” for a description of the currently supported private parameters.

R5-41 **[863]** If the SSP receives multiple `Furnish_AMA_Information` messages during the course of a call, only the information contained in the last message will be appended to the call record.

5.4 SSP non-call related messages

5.4.1 Termination_Notification message

The `Termination_Notification` message is sent in response to a `Send_Notification` message. When the SSP receives a `Send_Notification` message it continues on with normal call processing until the call is released. Once the call is released, the SSP will send a `Termination_Notification` message to the SCP. This message is described in section 5.8 of GR-1298-CORE and section 5.3.19 of GR-1299-CORE.

R5-42 **[687]** The SSP shall not send a `Termination_Notification` message when the `Send_Notification` component message is received in response to a LNP query.

R5-43 **[688]** The SSP shall not send a `Termination_Notification` message when the `Send_Notification` component is received as the first component in the package, or the first component is not call-related. This is a fatal application error.

R5-44 **[689]** If the SSP receives multiple `Send_Notification` requests for the same call and each request contains a unique TCAP transaction id, the SSP will send one `Termination_Notification` message for each request with a unique transaction id when an event occurs that releases the call.

R5-45 **[690]** The `Termination_Notification` message shall be populated with the parameters shown in Table 5-60.

Table 5-60 Parameters in the Termination_Notification message

Parameter name	Paragraph reference	
	This doc	GR-1299-CORE
EchoData	N/A	6.60
TerminationIndicator	N/A	6.128
ConnectTime	N/A	6.49
BusyCause	7.14	6.31
ExtensionParameter	7.30	6.68

R5-46 **[691]** The `EchoData` parameter is for the SCP to use to associate the SCP `Send_Notification` message with the SSP `Termination_Notification` message.

R5-47 **[692]** The `TerminationIndicator` parameter indicates the reason for sending the `Termination_Notification`. The following fields may be present inside the `TerminationIndicator` parameter: `AnswerIndication`, `BusyCauseIndication`, `UnrelatedErrorConditionIndication`, `ExceptionIndication`, `RerouteIndication`, `DisplayTextProvidedIndication`, and `NMControlListOverflowIndication`.

R5-48 **[693]** The `AnswerIndication` field shall be set to “yes” after the call is answered if an event occurs that releases the call.

R5-49 **[694]** The `BusyCauseIndication` field shall be set to “yes”, if after the SSP sends a Switch Call Related Message, the SSP detects that the call was not completed because of a `Network_Busy`, Busy Report, or Busy event or the caller abandons before the SSP receives

the Send_Notification Message and the T1 timer has not expired.

R5-50 **[695]** The UnrelatedErrorConditionIndication field shall be set to "yes" if the SSP detects an error that results in Final Treatment in a message except the Send_Notification message or if the call is cleared because it exceeded the serial trigger limit.

R5-51 **[696]** The ExceptionIndication field shall be set to "yes" if the SSP detects that the call was not completed AND an Exit Event Related to Incomplete calls, other than Network_Busy, Busy Reported, or Busy, is detected by the SSP.

R5-52 **[697]** The SSP shall set one, and only one of the previous four fields to "TRUE" for a given call.

R5-53 **[698]** The RerouteIndication field shall be set to "yes", if the SCP Response message sent with the Send_Notification message is Collect_Information, or Analyze_Route, and the SSP successfully rerouted the call based on the contents of the SCP response message.

R5-54 **[699]** The DisplayTextProvidedIndication field shall be set to "yes", if the SCP call related component sent with the Send_Notification message contained a DisplayText parameter and the SSP sent the contents of the DisplayText parameter to the access interface identified in the SCP Response message.

R5-55 **[700]** The NMControlListOverflowIndication field shall be set to "yes", if the ACG control list overflows.

R5-56 **[701]** The ConnectTime parameter indicates the call duration if the call was answered.

R5-57 **[702]** The ConnectTime parameter can only be present if the AnswerIndicator is set to "yes".

R5-58 **[703]** The BusyCause parameter specifies the reason the call could not be presented to the terminating party.

R5-59 **[704]** The BusyCause parameter can only be present if the BusyCauseIndicator is set to "yes".

R5-60 **[705]** The ExtensionParameter, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, "Extension parameters" for a description of the currently supported private parameters.

5.4.2 ACG_Overflow message

The ACG_Overflow message is sent by the SSP when a new ACG message is received by the SSP and there are no spaces available on the control list for the new control. The ACG_Overflow message will indicate what control

overflowed and whether the control was an SCP or SOCC control. This message is described in section 14 of GR-1298-CORE and section 5.34 of GR-1299-CORE.

R5-61 **[706]** The SSP shall send the `ACG_Overflow` message as a non-call related, unidirectional message sent to the SCP in an invoke last component.

R5-62 **[707]** If the package that contained the `ACG` control that overflowed contains a `Send_Notification` request, then the SSP shall set an ACG Overflow indicator in the `Termination_Notification` message (to be sent upon call completion). This is in addition to the `ACG_Overflow` message being sent.

R5-63 **[708]** The `ACG_Overflow` message shall be populated with the parameters shown in Table 5-61.

Table 5-61 Parameters in the ACG_Overflow message

Parameter name	Paragraph reference	
	This doc	GR-1299-CORE
ControlCauseIndicator	N/A	6.50
TranslationType	N/A	6.131
GlobalTitleAddress	N/A	6.84
ExtensionParameter	7.30	6.68

R5-64 **[709]** The `ControlCauseIndicator` parameter indicates whether the control is an SCP overload control or an SMS initiated control.

R5-65 **[710]** The `ControlCauseIndicator` parameter contains the number of digits to which the control is applied.

R5-66 **[711]** The `TranslationType` parameter is an integer from 0 to 255 indicating the translation type of the ACG control.

R5-67 **[712]** The `GlobalTitleAddress` parameter is a string of digits indicating the global title address of the ACG control.

R5-68 **[713]** The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, “Extension parameters” for a description of the currently supported private parameters.

5.4.3 ACG_Global_Ctrl_Restore_Success message

The `ACG_Global_Ctrl_Restore_Success` message is sent by the SSP when the ACG Global Restore Request has been successfully completed. This message is described in section 14 of GR-1298-CORE and section 5.3.3 of GR-1299-CORE.

R5-69 **[714]** The SSP shall send the `ACG_Global_Ctrl_Restore_Success` message as a non-call related response message in a return-result component.

R5-70 **[715]** The `ACG_Global_Ctrl_Restore_Success` message shall be populated with the parameters shown in Table 5-62.

Table 5-62 Parameters in the ACG_Global_Ctrl_Restore_Success message

Parameter name	Paragraph reference	
	This doc	GR-1299-CORE
ExtensionParameter	7.30	6.68

R5-71 **[716]** The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, “Extension parameters” for a description of the currently supported private parameters.

Chapter 6: Error messages

6.1 Introduction

This chapter provides an overview of the error messages supported by the NetworkBuilder implementation of Carrier AIN, together with rules for populating the messages with parameters. This chapter overrides portions of sections 4.5 and 7 of GR-1298-CORE. Encodings for all messages and parameters are provided in Appendix A, “Parameter encodings” or in Bellcore specification GR-1299-CORE.

Actions required on the part of the service switching point (SSP) or service control point (SCP) in generating or responding to these messages are described in GR-1298-CORE and GR-1299-CORE. Refer to the Bellcore documents for full requirements on message processing except where explicitly overridden in the following descriptions.

For each message described in this chapter, a table shows the expected parameters (in order) together with the nature of data to be provided in that parameter. Mandatory parameters are marked in **bold**. Parameters not listed are ignored (passed over) or unsupported (causes this parameter and the remainder of the message to be disregarded). Please refer to Table 2-17 for more information.

6.2 General requirements

The following requirements apply to all messages or all messages which carry the indicated parameter.

R6-1 **[71v2]** Processing of non-fatal TCAP protocol errors caused by unrecognized parameters in a message shall be handled using Alternative 1 (conditional requirements CR4-6 [94] through CR4-8 [96]) in section 4.2.1.4 of GR-1299-CORE.

R6-2 **[574]** Standard optional parameters that NetworkBuilder does not support and are not addressed in Table 2-16, Summary of parameters for error messages, shall be ignored and shall cause the remainder of the message to be discarded.

R6-3 **[72v4]** The following application errors detected in the SSP are considered fatal application errors:

- In an `Analyze_Route` message, no `ChargeNumber` parameter is provided, and no `ChargeNumber` was sent to the SCP by the SSP in the original query;
- In an `Analyze_Route` message, sent in response to a query originated from a TDP prior to `Info_Analyzed` which contains no routing parameters;
- In a `Send_To_Resource` or `Connect_To_Resource` message sent in a Response package with the `DisconnectFlag` parameter, the `ResourceType` is set to “play announcement and collect digits”;
- In a `Send_To_Resource` or `Connect_To_Resource` message sent in a Response package, the `DestinationAddress` parameter is included;
- In a `Send_To_Resource` or `Connect_To_Resource` message sent in a Conversation package, the `DisconnectFlag` parameter is included, except when the `ResourceType` is set to “play announcement”, in which case the non-fatal unexpected communication error is detected;
- In a `Send_To_Resource` or `Connect_To_Resource` message, the indicated tone or announcement is not datafilled in inswitch datafill on the UCS DMS-250 switch;
- Any message other than `Analyze_Route`, `Cancel_Resource_Event`, `Continue`, `Connect_To_Resource`, `Call_Info_To_Resource`, `Collect_Information`, `Send_To_Resource`, `Originate_Call`, `Merge_Call`, `Disconnect_Leg`, or `Acknowledge` is received in a Conversation package.

R6-4 **[73]** deleted

R6-5 **[74]** The SSP shall treat as a non-fatal application error the presence of an `ExtensionParameter` with an unrecognized or unsupported assignment authority. In this case, the `ExtensionParameter` shall be ignored.

R6-6 **[75v2]** The SSP shall treat as a non-fatal application error the presence within an `ExtensionParameter` of unrecognized or unsupported parameters. In this case, the SSP shall ignore the remainder of the `ExtensionParameter` and continue processing any remaining standard parameters.

R6-7 **[76]** The SSP shall treat as a non-fatal application error the presence within an `ExtensionParameter` of a parameter which is not applicable to the message. In this case, the SSP shall ignore the unrecognized or unsupported parameter, and proceed with the next parameter contained within the `ExtensionParameter`.

R6-8 **[77v2]** The SSP shall treat as a non-fatal protocol error any protocol error in the parameters contained within an `ExtensionParameter`. The SSP shall continue processing with the next parameter contained within the `ExtensionParameter`.

6.3 Application_Error message

The `Application_Error` message is sent by either the SSP or the SCP in response to errors detected in the AIN TCAP application logic. It is described in section 7 of GR-1298-CORE and section 5.4.1.1 of GR-1299-CORE.

R6-9 **[78]** The SSP shall report only fatal application errors. The SSP shall not maintain the TCAP transaction following a fatal application error.

R6-10 **[79]** The `Application_Error` message shall not be used to report errors in a closed transaction. The `Report_Error` message shall be used in this case.

R6-11 **[80]** After sending the `Application_Error` message, the SSP shall use default routing or treat the call, as indicated by switch datafill.

R6-12 **[81v2]** Receipt of an `Application_Error` message from the SCP in response to any call-related message sent by the SSP shall cause the call to proceed using default routing or treatment, as provided in switch datafill. The SSP shall not maintain the TCAP transaction.

R6-13 **[82]** The `Application_Error` message shall be populated with the parameters shown in Table 6-63.

Table 6-63 Parameters in the Application_Error message

Message name	Paragraph reference	
	This doc	GR-1299-CORE
ApplicationErrorString	7.12	6.25
ExtensionParameter	7.30	6.70

R6-14 **[83]** The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, “Extension parameters” for a description of the currently supported private parameters.

6.4 Close message

When used as an error message, the `Close` message is sent by the SCP to indicate a non-fatal Unexpected Communication error (see section 5.4 in GR-1298-CORE and section 4.3.3.3 in GR-1299-CORE). The SSP procedures for sending the `Close` message are described in section 4.5.5 of GR-1298-

CORE. Section 5.2.1.2 of GR-1299-CORE describes the format of the `Close` message.

It is recommended that the `UserID` and `BearerCapability` parameters not be used with the NetworkBuilder application when the `Close` message is sent from the SCP, as they will be ignored.

R6-15 **[84v2]** Receipt of the `Close` message from the SCP in response to any call-related message sent by the SSP shall cause the call to be given final treatment.

R6-16 **[365v3]** The SSP shall send the `Close` message to the SCP as the result of a non-fatal Unexpected Communication application error when receiving the `Request_Report_BCM_Event` message.

R6-17 **[575]** The following scenarios cause a `Close` message to be sent to the SCP only if EDPs are armed.

- The SSP performs an `Office_Code` query which results in applying treatment to the call.

R6-18 **[85]** The `Close` message shall be populated with the parameters shown in Table 6-64.

Table 6-64 Parameters in the Close message

Message name	Paragraph reference	
	This doc	GR-1299-CORE
UserID	7.49	6.134
BearerCapability	7.13	6.28
CloseCause	7.22	6.45
ExtensionParameter	7.30	6.70

R6-19 **[366]** The `UserID` parameter shall be filled as described in section 3.2.2.

R6-20 **[367]** The `BearerCapability` parameter shall be filled with the bearer capability of the agent as provisioned in the SSP when the `Close` message is sent.

R6-21 **[86]** The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, “Extension parameters” for a description of the currently supported private parameters.

6.5 Failure_Report message

The `Failure_Report` message is sent in response to another message when either SSP or SCP are unable to perform the operation requested. It is described in section 7 of GR-1298-CORE and section 5.4.1.3 of GR-1299-CORE.

R6-22 **[87]** The `Failure_Report` message shall be sent by the SSP only to indicate failure in processing a message (for example, resource failure); application and protocol errors shall not be reported with this message.

R6-23 **[88]** Receipt of a `Failure_Report` message from the SCP in response to any call-related message sent by the SSP shall cause the call to proceed as in the case of a fatal application error, unless it has successfully terminated on the outgoing agent. The SSP shall not maintain the TCAP transaction.

R6-24 **[89]** The `Failure_Report` message shall be populated with the parameters shown in Table 6-65.

Table 6-65 Parameters in the Failure_Report message

Message name	Paragraph reference	
	This doc	GR-1299-CORE
FailureCause	7.31	6.75
ExtensionParameter	7.30	6.70

R6-25 **[90]** The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, “Extension parameters” for a description of the currently supported private parameters.

6.6 Report_Error message

The `Report_Error` message is sent to indicate a fatal application error when that error is to be reported in a Unidirectional message. It is also used to report timeout of the T1 response message timer (no message was received when expected). It is described in section 7 of GR-1298-CORE and section 5.4.1.2 of GR-1299-CORE.

R6-26 **[91v2]** The `Report_Error` message shall be sent by the SSP only to indicate application errors in a closed transaction, or the expiration of the T1 message response timer. The `Report_Error` message shall be sent in a Unidirectional package, except in the case of a T1 timeout when the transaction is open, in which case a Response package shall be used.

R6-27 **[92v2]** Receipt of a `Report_Error` message from the SCP in response to any call-related message sent by the SSP shall cause the call to proceed as in the case of a fatal application error. The SSP shall not maintain the TCAP transaction.

R6-28 **[93]** The `Report_Error` message shall be populated with the parameters shown in Table 6-66.

Table 6-66 Parameters in the Report_Error message

Message name	Paragraph reference	
	This doc	GR-1299-CORE
ApplicationErrorString	7.12	6.25
ExtensionParameter	7.30	6.70

R6-29 **[94]** The `ExtensionParameter`, if present, shall consist of one or more customer- or vendor-specific parameters. See Chapter 8, “Extension parameters” for a description of the currently supported private parameters.

Chapter 7: AIN parameters

7.1 Introduction

This chapter provides specific information on the content of parameters supported by the NetworkBuilder implementation of Carrier AIN. This chapter overrides portions of GR-1298-CORE. Encodings for all messages and parameters are provided in Bellcore specification GR-1299-CORE, unless the encoding has been altered in this specification, in which case the encoding may be found in Appendix A, “Parameter encodings”.

Alterations of parameter encodings in this specification are limited to extending certain spare ranges for the values of selected fields in the parameter set, or removing limits on numbers of digits, enclosed sequences, and so forth

Certain messages consist of a sequence of fields which are themselves defined as parameters within this specification; in these cases, a table will indicate the permissible sub-parameters. Mandatory parameters and sub-parameters are indicated in **bold**.

7.2 General requirements

7.2.1 AINDigits parameters

The `AINDigits` parameter format is the basis for the digits parameters described in this section. This format is defined from the ISUP parameter descriptions in GR-394-CORE.

R7-1 **[95v2]** All digits parameters representing address or billing numbers shall be capable of containing 1–24 digits, except that the `AMADigitsDialedWC` parameter shall be capable of containing 3–27 digits. Note that digits shall be coded in BCD form.

R7-2 **[96]** The digits parameters representing billing numbers shall use an extended range of values for the Nature of Address field. Specific allowed values are noted in the parameter descriptions which follow. Encoding of all values is found in section A.2.2.

7.2.2 Trunk group numbers

The trunk group number format is used within the trunk group parameters (`PrimaryTrunkGroup`, and so forth) described in this section.

R7-3 **[97v2]** Trunk groups in routing parameters are identified by a 5-digit group number preceded by a 3-digit switch ID. The 5-digit number shall be unique for each trunk in a given switch, based on the group's value in table `CLLI`. Values in the range of 0–99999 are allowed, which exceeds the current UCS DMS-250 switch internal limits. The switch ID shall be unique for each switch in the network.

7.3 AccessCode parameter

The `AccessCode` parameter is described in section 6.1 of GR-1299-CORE.

R7-4 **[98v2]** The `AccessCode` parameter shall be used to encode an account code *only if the CAIN protocol version is* V2 *or lower.* The Nature of Address shall be set to `acct`. The Numbering Plan shall be set to `private`.

7.4 ACGGlobalOverride parameter

The `ACGGlobalOverride` parameter is described in section 6.3 of GR-1299-CORE.

R7-5 **[721]** The `ACGGlobalOverride` parameter re-initializes the ACG control list according to the value provided.

Valid `ACGGlobalOverride` values are shown in the following table:

Table 67 ACG Global Override Parameter Values

Enumerated Type	Enumerated Value
allItems	0
scpOverloadItems	1
smsInitCntrlExceptZeroGap	2
smsInitCntrl	3
not supported	4
not supported	5
craftInitCntrlExceptZeroGap	6
craftInitCntrl	7

Nortel Networks Proprietary and Confidential

7.5 AlternateCarrier parameter

The `AlternateCarrier` parameter is described in section 6.7 of GR-1299-CORE.

R7-6 **[368]** The SCP may send the `AlternateCarrier` parameter to the SSP to indicate a CIC to be used to determine an STS for subsequent translations and routing.

R7-7 **[369]** The `AlternateCarrier` parameter received from the SCP shall accept CIC values in the range of 3-4 digits.

7.6 AlternateTrunkGroup parameter

The `AlternateTrunkGroup` parameter is described in section 6.9 of GR-1299-CORE.

R7-8 **[99]** The `AlternateTrunkGroup` parameter shall be encoded with the switch ID and trunk group as detailed in section 7.2.2, with these eight digits filling the eight digit fields described in GR-1299-CORE.

7.7 AMADigitsDialedWC parameter

The `AMADigitsDialedWC` parameter is described in section 6.14 of GR-1299-CORE.

The `AMADigitsDialedWC` parameter provides digit strings which are to be entered into the DMS-250 Call Detail Record (CDR) for the call in progress. The parameter appears within an anonymous sequence of this parameter type within the SCP response messages which support it.

R7-9 **[267v2]** Up to six (6) `AMADigitsDialedWC` parameters may be sent in sequence within a response message.

Table 7-68 CDR field codes for AMADigitsDialedWC (Page 1 of 2)

Digits code	CDR field	
	Protocol Version <= v2	**Protocol Version >= v3**
001	PINDIGS	BILLNUM
002	ACCTCD	
003	BILLNUM	N/A
004	CIC	N/A
005	ORIGPRIVNUM	N/A
006	TERMPRIVNUM	N/A

Digits code	CDR field	
	Protocol Version <= v2	Protocol Version >= v3
301	PIN	
302	CIC	
303	ORIGPVN	
304	TERMPVN	
305	PROJECT CODE	

R7-10 **[268]** The AMADigitsDialedWC parameter shall contain 3–27 digits whose Nature of Address and Numbering Plan are both unknown. The first three digits of the string identify the field of the CDR into which the remaining digits are to be populated, as shown in Table 7-68. Excess digits beyond the limits of what fits in the CDR field shall be ignored.

R7-11 **[576]** When duplicate AMADigitsDialedWC digit tags are received, the digit string closest to the end of the message is retained and all other occurrences are discarded.

7.8 AMAAlternateBillingNumber parameter

The AMAAlternateBillingNumber parameter is described in section 6.10 of GR-1299-CORE.

R7-12 **[824]** The AMAAlternateBillingNumber parameter support is limited to the ability to receive the AMAAlternateBilling-Number parameter from the SCP and populate the ALTBILL CDR field. The ALTBILL CDR field can then be used in downstream processing.

R7-13 **[825]** (Deleted)

7.9 AMALineNumber parameter

The AMALineNumber parameter is described in section 6.15 of GR-1299-CORE.

The AMALineNumber parameter provides digit strings which are to be entered into the DMS-250 Call Detail Record (CDR) for the call in progress. The parameter appears within an anonymous sequence of this parameter type within the SCP response messages which support it.

R7-14 **[826]** Up to two (2) AMALineNumber parameters may be sent in sequence within a response message.

Table 7-69 CDR field codes for AMALineNumber parameter

Line Number Type	CDR Field
005	ANISP
006	CLGPTYNO, PRESIND

R7-15 **[827]** The AMALineNumber parameter shall contain 3–27 digits whose Nature of Address and Numbering Plan are both unknown. The first three digits of the string identify the field of the CDR into which the remaining digits are to be populated, as shown in Table 7-69. Excess digits beyond the limits of what fits in the CDR fields shall be ignored.

R7-16 **[828]** When both an AMALineNumber of type ANI (Line Number Type value of 006) and a ChargeNumber with a Nature of Address of ANI of the calling party; National number, ani, or i2ani are received, the AMALineNumber takes precedence in populating the ANISP CDR field.

R7-17 **[829]** The PRESIND field is mapped directly from the presentation restriction indicator field of the AMALineNumber parameter.

R7-18 **[830]** When both a CallingPartyID and an AMALineNumber of type Calling Party Number (Line Number Type value of 005) are available, precedence is given to the information contained in the AMALineNumber for CDR population.

R7-19 **[831]** When a sequence of two AMALineNumber parameters having the same Line Number Type value is received, the last one processed takes precedence in populating the corresponding CDR fields.

R7-20 **[832]** (Deleted)

7.10 AMAMeasure parameter

The AMAMeasure parameter is described in section 6.16 of GR-1299-CORE.

R7-21 **[577]** Following are the supported values for the AMAMeasure parameter:

Table 7-70 Supported AMAMeasure Values

AMAMeasure Value
connectTimeRecordedDestinationSSP

7.11 AMAslpID parameter

The AMAslpID parameter is described in section 6.21 of GR-1299-CORE.

R7-22 [722] The AMAslpID will be used to populate the SLPID CDR field.

R7-23 [723] The AMAslpID parameter is not checked for valid values.

7.12 ApplicationErrorString parameter

The ApplicationErrorString parameter is described in section 6.25 of GR-1299-CORE.

The ApplicationErrorString parameter contains fields which are defined as parameters themselves within this specification.

R7-24 [100] The ApplicationErrorString parameter shall be populated with the sub-parameters shown in Table 7-71.

Table 7-71 Sub-parameters of ApplicationErrorString

Parameter name	Paragraph reference	
	This doc	GR-1299-CORE
ErrorCause	N/A	6.69

R7-25 [370] (Deleted)

7.13 BearerCapability parameter

The BearerCapability parameter is described in section 6.28 of GR-1299-CORE.

R7-26 [101v3] The BearerCapability parameter shall be filled as indicated in GR-1299-CORE. *Only if the CAIN protocol version is* V2 *or lower* shall the following values which are noted as reserved shall be accepted values for the NetworkBuilder implementation:

- f7kHzAudio
- multiRate

If the above values are received and the *CAIN protocol version is* V3 *or higher,* then the value speech is sent.

R7-27 [371] For messages resulting from the Office_Code trigger, the BearerCapability parameter shall be restricted to only those values that are allowed for the AIN 0.1 specification.

7.14 BusyCause parameter

The `BusyCause` parameter is described in section 6.31 of GR-1299-CORE.

R7-28 **[269]** The `BusyCause` parameter shall use the `network-specific` Coding Standard and the extension bit in octet 2 set to indicate “diagnostic not included.” The General Location and 7-bit Cause Value fields shall be filled with the information returned to the SSP in the ISUP or PRI signalling which released the call with cause. Values outside of the range defined by GR-1299-CORE may be sent if they are defined on the UCS DMS-250 switch.

7.15 CalledPartyID parameter

The `CalledPartyID` parameter is described in section 6.33 of GR-1299-CORE.

R7-29 **[102v3]** The `CalledPartyID` parameter shall accept digit strings of 1–24 digits.

R7-30 **[578]** The Nature of Address values shown in Table 7-72 may be used *only if the CAIN protocol version is* V2 *or lower.*

Table 7-72 Nature of Address for CalledPartyID for CAIN protocol version V2 or lower

Nature of address	SSP Request	SCP Response
national	x	x
international	x	x
partitioned		x
vpn	x	x
hotline	x	
n00	x	

The Numbering Plan shall be `ISDN` for the `national`, `international`, and `partitioned` Nature of Address values, and `private` otherwise.

R7-31 **[579]** The Nature of Address values shown in Table 7-73 may be used *only if the CAIN protocol version is* V3 *or higher.*

Table 7-73 Nature of Address for CalledPartyID for CAIN protocol version V3 or higher

Nature of address	SSP Request	SCP Response
national	x	x
international	x	x
partitioned		x
vpn		x
subscriber number	x	
unknown	x	

The Numbering Plan shall be `ISDN` for the `national`, `international`, and `partitioned` Nature of Address values, and `private` otherwise.

7.16 CallingPartyID parameter

The `CallingPartyID` parameter is described in section 6.36 of GR-1299-CORE.

R7-32 **[270v2]** The `CallingPartyID` parameter shall accept digit strings of 1–24 digits.

R7-33 **[103v2]** (Deleted)

R7-34 **[372v2]** Table 7-74 summarizes the Nature of Addresses that shall be supported for SCP response messages.

Table 7-74 Supported Nature of Address Values for CallingPartyID in SCP response messages (page 1 of 2)

Nature of address
unknown
unique subscriber number
unique significant national number
unique international number
vpn

Nature of address
non-unique subscriber number
non-unique national number
non-unique international number
test line test code

See section 3.2.4 for a further discussion of calling party identification.

7.17 Carrier parameter

The `Carrier` parameter is described in section 6.37 of GR-1299-CORE.

R7-35 **[104v3]** The `Carrier` parameter shall be used to transmit the carrier identification code (CIC) received on FGD or SS7 trunks, or as provisioned in switch datafill.

R7-36 **[271]** If the CIC was derived from switch datafill, the Carrier Selection Indicator shall be set to “presubscribed and not input”; otherwise, the received indication shall be sent to the SCP. In cases where no indication is received, the Carrier Selection Indicator shall be set to “no indication”.

R7-37 **[373]** The SCP may send the `Carrier` parameter to the SSP to indicate a CIC to be used to determine an STS for subsequent translations and routing.

R7-38 **[374]** The `Carrier` parameter received from the SCP shall accept CIC values in the range of 3-4 digits.

7.18 CcID parameter

The `CcID` parameter is described in section 6.40 of GR-1299-CORE.

R7-39 **[580v2]** Following are the supported values for the `CcID` parameter:

Table 7-75 Supported CcID Values (page 1 of 2)

CcID Value
originatingSetup
stable2Party
terminatingSetup
threePartySetup
threePartySetupComplement
partyOnHold

CcID Value
partyOnHoldComplement
callWaiting
callWaitingComplement
stableMParty
transfer
forward

7.19 ChargeNumber parameter

The `ChargeNumber` parameter is described in section 6.41 of GR-1299-CORE.

R7-40 **[105v4]** The `ChargeNumber` parameter received from the SCP shall contain digit strings of 1–24 digits. *If the CAIN protocol version is* V3 *or lower,* only the Nature of Address values shown in Table 7-76 are supported in a `ChargeNumber` received from the SCP. Receipt of an unsupported Nature of Address from the SCP will be treated as a non-fatal application error.

Table 7-76 SCP ChargeNumber Nature of Address: protocol version V3 or lower

Received Nature of Address value
subscriber number
ani
i2ani
authcode
mccs
n00
ANI of the calling party; national number

R7-41 **[725v2]** *If the CAIN protocol version is* V3 *or lower,* the Numbering Plan of the `ChargeNumber` parameter sent from the SSP shall be `ISDN` or `private`.

R7-42 **[726v2]** The `ChargeNumber` parameter sent by the SSP shall consist of digit strings of 1–24 digits. The Nature of Address values shown in Table 7-77 may be used *depending on the CAIN protocol version.*

Table 7-77 SSP Nature of Address

Nature of address	CAIN Protocol Version	
	V3 or lower	V4 or higher
i2ani	x	x[a]
authcode	x	x[a]
mccs	x	x[a]
n00	x	x[a]
international	x	x[a]
unknown	x	x[a]
ANI of the calling party; National number	x	x
ANI of the calling party; Subscriber number		x
ANI not available or not provided		x
ANI of the called party included; Subscriber number		x
ANI of the called party; not included		x
ANI of the called party included; National number		x
Other		x[a]

a. This value may only be sent after being received in an incoming IAM. See requirement R7-43.

R7-43 **[866]** *If the CAIN protocol version is* V4 *or higher*, then a Charge Number parameter received in an IAM for an originating SS7 EANT or IMT will be sent to the SSP without validation.

R7-44 **[375]** The `ChargeNumber` Nature of Address shall be restricted to the following values in queries resulting from the `Office_Code` trigger:

- `ANI of the calling party; National number`
- `international`
- `unknown`

R7-45 **[376v2]** An ANI sent in the `ChargeNumber` parameter shall have a Nature of Address of `ANI of the calling party;`

`National number` *only if the CAIN protocol version is* V2 *or higher.* This only applies to the `ChargeNumber` parameter sent from the SSP.

7.20 ChargePartyStationType parameter

The `ChargePartyStationType` parameter is described in section 6.42 of GR-1299-CORE.

R7-46 **[377]** The `ChargePartyStationType` parameter shall be filled with the Information Digits from the call.

R7-47 **[378]** (Deleted)

R7-48 **[581v2]** (Deleted)

7.21 ClearCause parameter

The `ClearCause` parameter is described in section 6.43 of GR-1299-CORE. Please refer to Appendix C, “ClearCause meaning” for more information.

R7-49 **[379]** The following values are supported:

Table 7-78 Supported ClearCause Values (page 1 of 2)

ClearCause Value
normal
timeout
resourceCancelled
userAbandon
invalidCode
failure
channelsBusy
calledPartyAnswered
resourceNotAvailable
resourceTypeNotSupported
taskRefused
protocolError
abort
suppServiceInvoked

ClearCause Value
strCancelled

7.22 CloseCause parameter

The `CloseCause` parameter is described in section 6.45 of GR-1299-CORE.

R7-50 **[380]** Deleted.

7.23 CollectedAddressInfo parameter

The `CollectedAddressInfo` parameter is described in section 6.46 of GR-1299-CORE.

R7-51 **[106v3]** The `CollectedAddressInfo` parameter shall accept digit strings of 1–24 digits. If the version designation of the CAIN protocol version is V0 or V1, then the Nature of Address and Numbering Plan indicators shall both be set to `unknown`. If the version designation of the CAIN protocol version is V2 or higher, then the Nature of Address and Numbering Plan shall be filled as per GR-1298-CORE requirements 189, 190, and 191; and GR-1299-CORE [207] and [311].

7.24 CollectedDigits parameter

The `CollectedDigits` parameter is described in section 6.47 of GR-1299-CORE.

R7-52 **[107v3]** On SSP queries, the Nature of Address field shall be set to `pin` and the Numbering Plan to `private` *only if the CAIN protocol version is* V2 *or lower.*

R7-53 **[272]** When used in the `Resource_Clear` message, the `CollectedDigits` parameter shall be encoded as specified in GR-1299-CORE, and represents an undifferentiated string of digits whose nature is not known to the SSP. The Nature of Address and Numbering Plan indicators shall both be set to `unknown` in this case.

7.25 CsID parameter

The `CsID` parameter is described in section 6.53 of GR-1299-CORE.

R7-54 **[843]** The following values are supported.

Table 7-79 Supported CsID Values (page 1 of 2)

CsID Value
1

CsID Value
2

7.26 DestinationAddress parameter

The `DestinationAddress` parameter is described in section 6.55 of GR-1299-CORE.

R7-55 **[273v2]** The `DestinationAddress` parameter shall be encoded as described in GR-1299-CORE, except that the Nature of Address may be `vpn, international` as well as `national`.

R7-56 **[582]** The Numbering Plan shall be `private` when the Nature of Address is `vpn`, and `ISDN` otherwise.

R7-57 **[583]** The `DestinationAddress` may contain between 1 and 24 digits inclusively.

7.27 DisplayText parameter

The `DisplayText` parameter is described in section 6.58 of GR-1299-CORE.

R7-58 **[584]** Table 7-80 lists the supported values for the `DisplayInformation` specifier within the `DisplayText` parameter. A value other than those indicated is ignored.

Table 7-80 Supported DisplayInformation Specifier Values

DisplayInformation Specifier
callingPartyName
originalCalledName
redirectingName

R7-59 **[585]** When duplicate `DisplayInformation` specifiers are received, the string closest to the end of message is retained and all other strings are discarded.

7.28 EDPNotification parameter

The `EDPNotification` parameter is described in section 6.62 of GR-1299-CORE.

R7-60 **[586v2]** CAIN treats the `oDisconnect` bit as NOT “always ‘0’”, unlike Bellcore. Refer to A.2.8, “Encoding the EDPNotification parameter” in this document.

7.29 EDPRequest parameter

The EDPRequest parameter is described in section 6.63 of GR-1299-CORE.

R7-61 **[587v2]** (Deleted)

7.30 ExtensionParameter parameter

The ExtensionParameter parameter is described in section 6.68 of GR-1299-CORE.

R7-62 **[108]** The ExtensionParameter parameter shall be encoded as described in GR-1299-CORE. Note that it shall always be encoded as a constructor parameter, as it is an ASN.1 SEQUENCE.

R7-63 **[109]** The value of the Assignment Authority field shall be the ASN.1 object identifier for this document as (see section 2.5.4). Encoding of this identifier is shown in section A.2.4.

R7-64 **[110]** The extension parameter may contain parameters from the standard extended parameter set as described in Chapter 8, “Extension parameters”. Bellcore-specified AIN parameters may not be encapsulated within this parameter.

7.31 FailureCause parameter

The FailureCause parameter is described in section 6.73 of GR-1299-CORE.

R7-65 **[381v2]** The following values are supported:

Table 7-81 Supported FailureCause Values

FailureCause values
applicationError
unavailableResources

7.32 FeatureActivatorID parameter

The FeatureActivatorID parameter is described in section 6.75 of GR-1299-CORE.

R7-66 **[274]** When used in the context of the O_Feature_Requested message, the FeatureActivatorID parameter shall always be encoded in one octet, and shall contain the ID number of the feature

which was specified in the OFTRREQ trigger table's FEAT selector. The defined values are shown in Table 7-82.

Table 7-82 Feature codes for FeatureActivatorID

ID value	Feature indicated
1	CARD
2	AUTH
3	ADDR

7.33 ForwardCallIndicator parameter

The ForwardCallIndicator parameter is described in section 6.76 of GR-1299-CORE.

R7-67 **[382]** The ForwardCallIndicator (FCI) parameter shall indicate that a Local Number Portability (LNP) check has been performed at the SCP by setting bit M of the FCI.

R7-68 **[383]** The values of the other fields in the FCI shall be ignored in this release.

7.34 GenericAddress parameter

The GenericAddress parameter is described in section 6.79 of GR-1299-CORE.

R7-69 **[111v2]** The GenericAddress parameter shall be encoded as described in GR-1299-CORE, except that spare Type of Digits values shall be allowed to convey addresses for the SSP to interpret. When one of these spare values is used, the SSP shall not directly transfer the generic address to the outgoing call's signalling (for example, the ISUP IAM). Instead, the address shall be used within the SSP and only sent in call processing signalling messages if required by the feature description involved.

R7-70 **[112v2]** The following address types are defined (see section A.2.6 for the encodings):

- AlternateOutpulseNo
- SecondAlternateOutpulseNo
- OverflowRoutingNo
- LNPPortedDialedNo
- DNIS

Nortel Networks Proprietary and Confidential

R7-71 **[384]** The values of the LNP Ported Dialed Number and DNIS GenericAddress parameters are copied directly into the outgoing ISUP IAM message.

R7-72 **[113]** The Nature of Address and Encoding Scheme values for the outpulse number addresses shall be the same as for the addresses in the standard `OutpulseNumber` parameter. (see section 7.40).

R7-73 **[114v2]** The Nature of Address and Encoding Scheme values for the routing number address, the LNP Ported Dialed Number, and the DNIS number shall be the same as for the addresses in the standard `CalledPartyID` parameter.

7.35 GenericAddressList parameter

The `GenericAddressList` parameter is described in section 6.80 of GR-1299-CORE.

R7-74 **[115v2]** The `GenericAddressList` parameter shall be encoded as specified in GR-1299-CORE. Up to five (5) contained `GenericAddress` parameters are supported.

7.36 IPReturnBlock parameter

The `IPReturnBlock` parameter is described in section 6.86 of GR-1299-CORE.

R7-75 **[588]** The `IPReturnBlock` may be sent containing information the SCP requested from a Virtual IP. Refer to section A.7 for the encoding for the `IPReturnBlock` when used in this manner.

R7-76 **[589]** The maximum number of bytes the `IPReturnBlock` shall contain is summarized in Table 7-83.

Table 7-83 Size restrictions for the IPReturnBlock (page 1 of 2)

Messaging Scenario	Parameter Breakdown	IP Trunk Type	
		PRI[a]	ISUP
Resource_Clear	ParmID and length byte overhead	3	3
	Payload	92	115
	Total	95	118

Messaging Scenario	Parameter Breakdown	IP Trunk Type	
		PRI[a]	ISUP
Call_Info_From_Resource	ParmID and length byte overhead	3	3
	Payload	92	115
	Total	95[b,c]	118
CTR_Clear	ParmID and length byte overhead	3	3
	Payload	92	115
	Total	95	118

a. Requires XPM load ELI81AZ or later.
b. Due to CM-XPM interactions, the Resource_Clear message is not sent when this limit is exceeded.
c. This requires CCM10, XPM10 feature AJ5132 - *PRI LARGE FIE FOR INCOMING FACILITY MSG* which increased the size of an incoming FIE from 57 bytes to 113 bytes.

7.37 JurisdictionInformation parameter

The `JurisdictionInformation` parameter is described in section 6.87 of GR-1299-CORE.

R7-77 **[590]** Deleted.

R7-78 **[591]** Deleted.

R7-79 **[592]** Deleted.

7.38 Lata parameter

The `Lata` parameter is described in section 6.89 of GR-1299-CORE.

R7-80 **[833]** The `Lata` parameter is supported by the following trunk group types: DAL, EANT, and AXXESS.

R7-81 **[834]** Support is provided only when the value of CAIN_PROTOCOL_STREAM in table CAINPARM is UCS11 or higher.

R7-82 **[835]** If the parameter is included in a TDP-R message from the SSP, the value of the `Lata` parameter is captured in the LATA field of the CDR.

7.39 LegID Parameter

The `LegID` parameter is described in section 6.90 of GR-1299-CORE.

R7-83 [593v2] The following values are supported for the LegID parameter.

Table 7-84 Supported values for the LegID parameter

LegID Value
0
1
2

7.40 OutpulseNumber parameter

The `OutpulseNumber` parameter is described in section 6.98 of GR-1299-CORE.

R7-84 [116v2] The `OutpulseNumber` parameter shall accept digit strings of 1–24 digits. The Nature of Address values shown in Table 7-85 may be used.

Table 7-85 Supported incoming Nature of Address values for OutpulseNumber

Nature of address
Not Applicable
Subscriber number
Significant National number
International number
partitioned
vpn
Subscriber number, operator requested (0+ call)
National number, operator requested (0+ call)
International number, operator requested (0+ call)
No address present, operator requested (0-, 10XXX+0(0), or 00- call)
No address present, cut-through call to carrier

The Numbering Plan in incoming `OutpulseNumber` parameters is ignored.

R7-85 **[727]** An incoming `OutpulseNumber` with an NOA of `partitioned` is considered to be an `international` number with Numbering Plan set to `ISDN`.

R7-86 **[728]** An incoming `OutpulseNumber` with an NOA of `vpn` is considered to be a `National` number with Numbering Plan set to `ISDN`.

R7-87 **[729]** An incoming `OutpulseNumber` with an NOA out of the accepted range is considered to be a `National` number with Numbering Plan set to `ISDN` and a log is generated.

R7-88 **[117]** The digits of the `OutpulseNumber` parameter shall never be used as an address for in-switch translations or routing.

7.41 PointInCall parameter

The `PointInCall` parameter is described in section 6.101 of GR-1299-CORE.

R7-89 **[385v2]** The following values are supported:

Table 7-86 Supported PointInCall Values

PointInCall values
collectInformation
sendCall
oAlerting
oActive
oSuspended

7.42 PrimaryTrunkGroup parameter

The `PrimaryTrunkGroup` parameter is described in section 6.104 of GR-1299-CORE.

R7-90 **[118]** The `PrimaryTrunkGroup` parameter shall be encoded with the switch ID and trunk group as detailed in section 7.2.2, with these eight digits filling the eight digit fields described in GR-1299-CORE.

7.43 ResourceType parameter

The `ResourceType` parameter is described in section 6.108 of GR-1299-CORE.

R7-91 **[119v3]** The `ResourceType` parameter shall be filled as specified in GR-1299-CORE, except that only the values “play announcement”,

"play announcement and collect digits", and "flex parameter block" shall be used.

7.44 SecondAlternateCarrier parameter

The `SecondAlternateCarrier` parameter is described in section 6.114 of GR-1299-CORE.

R7-92 **[386]** The SCP may send the `SecondAlternateCarrier` parameter to the SSP to indicate a CIC to be used to determine an STS for subsequent translations and routing.

R7-93 **[387]** The `SecondAlternateCarrier` parameter received from the SCP shall accept CIC values in the range of 3-4 digits.

7.45 SecondAlternateTrunkGroup parameter

The `SecondAlternateTrunkGroup` parameter is described in section 6.115 of GR-1299-CORE.

R7-94 **[120]** The `SecondAlternateTrunkGroup` parameter shall be encoded with the switch ID and trunk group as detailed in section 7.2.2, with these eight digits filling the eight digit fields described in GR-1299-CORE.

7.46 STRConnection parameter

R7-95 **[594]** (Deleted)

7.47 StrParameterBlock parameter

The `StrParameterBlock` parameter is described in section 6.124 of GR-1299-CORE.

R7-96 **[121]** (Deleted)

R7-97 **[275]** Within the `MaximumDigits` field of the `AnnouncementDigitBlock`, spare values shall be used to indicate a maximum number of digits to collect when collecting a variable number of digits. Encoding of these spare values is shown in section A.2.7.

R7-98 **[595]** The absence of a `DestinationAddress` parameter and a `StrParameterBlock` encoded as a Flex parameter block indicates that the `StrParameterBlock` is destined for a Virtual IP.

R7-99 **[596v2]** The maximum number of bytes the `StrParameterBlock` shall contain when it is encoded as a `FlexParameterBlock` is summarized in Table 7-87.

Table 7-87 Size restrictions for the StrParameterBlock (FlexParameterBlock)

Messaging Scenario	Parameter Breakdown	IP Trunk Type	
		PRI[a]	ISUP
Send_To_Resource	ParmID, tag, and length bytes overhead (for both STRParameterBlock and FlexParameterBlock)	5	5
	Payload of FlexParameterBlock	86	111
	Total	91	116
Call_Info_To_Resource	ParmID, tag, and length bytes overhead (for both STRParameterBlock and FlexParameterBlock)	5	5
	Payload of FlexParameterBlock	86	111
	Total	91	116
Connect_To_Resource	ParmID, tag, and length bytes overhead (for both STRParameterBlock and FlexParameterBlock)	5	5
	Payload of FlexParameterBlock	86	111
	Total	91	116

a. Requires XPM load ELI81AZ or later.

7.48 TriggerCriteriaType parameter

The `TriggerCriteriaType` parameter is described in section 6.133 of GR-1299-CORE.

R7-100 [122v3] The `TriggerCriteriaType` parameter shall be filled as shown in Table 7-88

Table 7-88 TriggerCriteriaType values (page 1 of 2)

Trigger	Criteria	Value
Offhook Immediate	N/A	offHookImmediate
O Feature Requested	N/A	oFeatureActivator
OffHook Delay	N/A	offHookDelay
Shared Interoffice Trunk	ANI	sio_ani[a]
	ADIN	sio_adin[a]
	national/VPN address	sio_addr[a]
	II digits	sio_info[a]
	CIC code	sio_cic[a]
	N00 address	sio_n00[a]
	international address	sio_intl[a]
	standard value	sharedIOTrunk
PRI B-Channel (Channel Setup PRI)	ADIN	csp_adin[a]
	CLID	csp_clid[a]
	national/VPN address	csp_addr[a]
	N00 address	csp_n00[a]
	international address	csp_intl[a]
	standard value	channelSetupPRI
Specific Feature Code	N/A	specificFeatureCode
Custom Dialing Plan	VPN address	customIntercom

Trigger	Criteria	Value
Specific Digit String	national/VPN address	npaNXXXXXX
	N00 address	sds_n00[a]
	ANI	sds_ani[a]
	ADIN	sds_adin[a]
	international address	countryCodeNPANXXXXXX
	II digits	sds_info[a]
	CIC code	sds_cic[a]
	standard value	varies according to GR-1298-CORE requirement [75]
Office Code[b]	national	npa
	national	NPANXX
	national	NPANXXXXXX
	national	NPANXXX
	national	NPANXXXX
	national	NPANXXXXX
	national	lnpOfficeCode
Network Busy	N/A	networkBusy
O Called Party Busy	N/A	oCalledPartyBusy
O No Answer	N/A	oNoAnswer
O IEC Reorigination	N/A	oIECReorigination[a]
Termination Attempt	N/A	terminationAttempt

a. Spare values in GR-1299-CORE; see section A.2.5 for the encoding of these values.
b. The actual value sent is provisioned in inswitch datafill as one of the possible values.

7.49 UserID parameter

The `UserID` parameter is described in section 6.134 of GR-1299-CORE.

R7-101 [123v4] *When the CAIN protocol version is* v2 *or lower*, the `UserID` parameter shall take the DN format as described in GR-1299-CORE for all triggers excluding `Office_Code` and `Tollfree_Services`.

R7-102 [124v3] The switch ID and trunk group are encoded into ten digits as follows *only if the CAIN protocol version is* V2 *or lower*:

- the first two digits shall be zero
- the third through fifth digits shall be the switch ID

- the remaining five digits shall be the trunk group number, padded with leading zero digits as required

R7-103 [388] In messages sent as a result of the `Office_Code` trigger, the `UserID` parameter shall take the `TrunkGroupID` form as described in GR-1299-CORE.

R7-104 [730v2] *When the CAIN protocol version is* V3 *or greater and PRIVATE_FACILITY_GROUP_USERID in table CAINPARM is set to N*, the `UserID` parameter shall take the `TrunkGroupID` form in all messages sent by the SSP.

R7-105 [864] *When the CAIN protocol version is* V3 *or greater and PRIVATE_FACILITY_GROUP_USERID in table CAINPARM is set to Y*, the `UserID` parameter shall take the `PrivateFacilityGID` form as described in GR-1299-CORE in all messages sent by the SSP.

See section 3.2.2 for a further description of user identification.

7.50 VerticalServiceCode parameter

The `VerticalServiceCode` parameter is described in section 6.135 of GR-1299-CORE.

R7-106 [276v2] The `VerticalServiceCode` parameter shall be used to transmit feature codes dialed by a subscriber. It shall be encoded as specified in GR-1299-CORE, except that the first digit may be "#" as well as "*" or a digit 0–9. Up to 10 digits may be included in this parameter. The Nature of Address shall be set to `hotline` and the Numbering Plan indicator shall be set to `private`.

R7-107 [389] This parameter is supported for the `Info_Analyzed` message *only if the CAIN protocol stream is* UCS06 *or higher*

Chapter 8: Extension parameters

8.1 Introduction

This chapter provides an overview of the standard extended parameter set supported by the NetworkBuilder implementation of Carrier AIN. The parameters described in this chapter are intended to be sent within the AIN `ExtensionParameter` parameter. Encodings for these parameters are provided in Appendix A, “Parameter encodings”.

8.2 General requirements

The following requirements apply to all parameters described in this chapter.

R8-1 **[125]** The extension parameters shall only be transmitted within the AIN `ExtensionParameter` parameter in an AIN message. They shall not be transmitted within the body of an AIN message nor within any other AIN parameter.

R8-2 **[126]** Absence of the `ExtensionParameter` parameter, or of any specific extension parameter within it, shall not constitute a reportable application or protocol error.

R8-3 **[127]** The extension parameters shall be ancillary to the basic AIN functionality. The service switching point (SSP) shall be capable of processing the two-party call, including validation of fundamental billing data, in the absence of extension parameters in the service control point (SCP) response to a query. However, it is understood that certain feature functionality which extends the basic two-party call may not operate fully without the data that the extension parameters provide.

R8-4 **[128]** The SSP shall maintain default information in-switch for the contents of all extension parameters. In the event that an extension parameter is not received, the default information shall be used in its place. It is expected that these default values will restrict the functionality of certain feature extensions to the basic two-party call.

8.2.1 Compatibility parameters

Several of the extension parameters described here are intended to provide parity with respect to similar IN/1 features that exist today in the UCS DMS-250 switch. One feature of IN/1 which is not provided here regards the

collection and/or validation of account code or PIN digits. No digit collection shall be performed as the result of processing any of these parameters.

8.3 Summary of extension parameter identifiers

Table 8-89 and Table 8-90 summarize the extension parameter set.

Table 8-91, Table 8-92 and Table 8-93 summarize the applicability of the extension parameters to specific messages.

Table 8-89 `NT_DMS250_EXTENSION_PARAMETER_FAMILY` extension parameter summary (page 1 of 2)

Name	Identifier	Value in hex
servTranslationScheme	1	81
callType	2	82
satRestriction	3	83
classOfSvc	4	84
callBranding	5	85
busyRoute	8	A8
billSequenceNumber	9	89
universalAccess	10	8A
connectToSCU	11	8B
cainGroup	12	8C
adin	13	8D
origTrunkInfo	14	AE
treatment	15	8F
reorigCall	16	90
termTrunkInfo	17	B1
reorigAllowed	18	92
networkBusyActions	19	93
oCalledPartyBusyActions	20	94
oNoAnswerActions	21	95
edpBuffer	22	96
univIdx	23	97
netinfo	24	98
t1Overflow	25	99
lnpReceived	26	9A
callCtrl	27	9B
subscriptionInfo	28	9C
connectTime	29	9D

Table 8-89 `NT_DMS250_EXTENSION_PARAMETER_FAMILY` extension parameter summary (page 1 of 2)

Name	Identifier	Value in hex
strConnectionType	30	9E
pretranslatorName	31	9F 1F
primaryTrunkGroupSTS	32	9F 20
alternateTrunkGroupSTS	33	9F 21
secondAlternateTrunkGroupSTS	34	9F 22
overflowRoutingNoSTS	35	9F 23
numReorig	36	9F 24
amaDigits	37	9F 25
cainPRT	38	9F 26
jurisdictionInformation	39	9F 27
collectedAddress	40	9F 28
switchID	41	9F 29
billingNumber	42	9F 2A
accountCode	43	9F 2B
pinDigits	44	9F 2C
acgRequery	45	9F 2D
shfelegs	46	9F 2E

Table 8-90 `infoCollectedExtensionAa` extension parameter summary

Name	Identifier	Value in hex
accessType	5	85

Table 8-91 `NT_DMS250_EXTENSION_PARAMETER_FAMILY` extension parameter to SSP message mapping (page 1 of 2)

Parameter name (x = allowed)	**Info_Analyzed**	**Info_Collected**	**Network_Busy**	**Origination_Attempt**	**O_Answer**	**O_Called_Party_Busy**	**O_Feature_Requested**	**O_No_Answer**	**O_Term_Seized**	**O_Mid_Call**	**Termination_Attempt**	**Timeout**	**O_Disconnect**	**Close**	**Termination_Notification**	**ACG_Overflow**	**ACG_Global_Ctrl_Restore_Success**
busyRoute			x			x		x									
universalAccess	x	x	x			x	x	x		x							
cainGroup	x	x	x	x		x	x	x		x	x						
adin	x	x		x			x										
origTrunkInfo	x	x	x	x		x	x	x		x	x						
treatment	x	x					x			x							
reorigCall	x	x					x				x						
termTrunkInfo			x		x	x		x	x		x	x	x				
univIdx	x	x					x										
netinfo	x	x	x			x	x	x		x							
t1Overflow	x	x	x	x		x	x	x			x						
lnpReceived	x	x					x										
subscriptionInfo	x	x	x	x		x	x	x		x	x						
cainPRT							x										
connectTime												x	x				
numReorig										x							
jurisdictionInformation		x	x	x		x	x	x		x							
collectedAddress			x			x		x									
switchID	x	x	x	x			x	x		x	x						
billingNumber	x	x	x	x			x	x		x	x						
accountCode	x	x					x			x							
pindigits	x	x					x										

Parameter name (x = allowed)	**Info_Analyzed**	**Info_Collected**	**Network_Busy**	**Origination_Attempt**	**O_Answer**	**O_Called_Party_Busy**	**O_Feature_Requested**	**O_No_Answer**	**O_Term_Seized**	**O_Mid_Call**	**Termination_Attempt**	**Timeout**	**O_Disconnect**	**Close**	**Termination_Notification**	**ACG_Overflow**	**ACG_Global_Ctrl_Restore_Success**
acgRequery	x	x	x	x		x	x	x		x	x						

Table 8-92 `infoCollectedExtensionAa` extension parameter to SSP message mapping

Parameter name (x = allowed)	Info_Analyzed	Info_Collected	Network_Busy	Origination_Attempt	O_Answer	O_Called_Party_Busy	O_Feature_Requested	O_No_Answer	O_Term_Seized	O_Mid_Call	Termination_Attempt	Timeout	O_Disconnect	Close	Termination_Notification	ACG_Overflow	ACG_Global_Ctrl_Restore_Success
accessType		x															

Table 8-93 `NT_DMS250_EXTENSION_PARAMETER_FAMILY` extension parameter to SCP message mapping (page 1 of 2)

Parameter name (x = allowed)	Analyze_Route	Cancel_Resource_Event	Continue	Disconnect	Send_To_Resource	Request_Report_BCM_Event	Connect_To_Resource	Collect_Information	Authorize_Termination	Call_Info_To_Resource	Send_Notification	ACG	ACG_Global_Ctrl_Restore	Furnish_AMA_Information	Originate_Call	Merge_Call	Disconnect_Leg	Acknowledge
servTranslationScheme	x				x		x											
callType	x																	
satRestriction	x																	
classOfSvc	x																	
callBranding	x																	
billSequenceNumber	x		x	x	x		x		x									
connectToSCU			x															
cainGroup	x							x										
treatment				x	x		x											
reorigAllowed	x																	
networkBusyActions						x												

Parameter name (x = allowed)	**Analyze_Route**	**Cancel_Resource_Event**	**Continue**	**Disconnect**	**Send_To_Resource**	**Request_Report_BCM_Event**	**Connect_To_Resource**	**Collect_Information**	**Authorize_Termination**	**Call_Info_To_Resource**	**Send_Notification**	**ACG**	**ACG_Global_Ctrl_Restore**	**Furnish_AMA_Information**	**Originate_Call**	**Merge_Call**	**Disconnect_Leg**	**Acknowledge**
oCalledPartyBusyActions						x												
oNoAnswerActions						x												
edpBuffer						x												
univIdx	x				x		x											
netinfo	x																	
callCtrl	x		x															
pretranslatorName					x		x											
primaryTrunkGroupSTS	x																	
alternateTrunkGroupSTS	x																	
secondAlternateTrunkGroupSTS	x																	
overflowRoutingNoSTS	x																	
amaDigits										x								
strConnectionType					x		x											
billingNumber	x		x		x		x											
accountCode	x		x		x		x											
pinDigits	x		x		x		x											
shfelegs						x												

8.4 DMS-250 routing feature compatibility

The following extension parameters are provided to offer compatibility with UCS DMS-250 switch translation and routing features.

8.4.1 STS and OPART/TPART pairs

Five serving translation scheme (STS) extension parameters are defined. Three (`primaryTrunkGroupSTS`, `alternateTrunkGroupSTS`,

secondAlternateTrunkGroupSTS) directly correspond to direct routing standard parameters (PrimaryTrunkGroup, AlternateTrunkGroup, and SecondAlternateTrunkGroup). One (overflowRoutingNoSTS) corresponds to the OverflowRoutingNo in the GenericAddressList. The last (servTranslationScheme) is used with the CalledPartyID and with other standard parameters when their specific STS is not present in the Analyze_Route message.

R8-5 **[129v2]** The servTranslationScheme extension parameter shall contain the value of the STS as indicated by the SCP. It shall only be returned from the SCP within the Analyze_Route, Connect_To_Resource or Send_To_Resource messages.

R8-6 **[597]** The primaryTrunkGroupSTS, alternateTrunkGroupSTS, secondAlternateTrunkGroupSTS, and overflowRoutingNoSTS extension parameters shall contain the value of the STS as indicated by the SCP. It shall only be returned from the SCP within the Analyze_Route message.

R8-7 **[130v2]** The STS in use for the call shall be used any time (after its receipt by the SSP) where the SSP would perform an internal translation of a routing number. The call detail record (CDR) shall be populated with an OPART and TPART derived from the STS.

R8-8 **[598]** (Deleted)

R8-9 **[277v2]** If the servTranslationScheme is returned in a Send_To_Resource or Connect_To_Resource message, it shall only be used to translate the address in an accompanying DestinationAddress parameter in order to route the call to an IP.

R8-10 **[131]** Encoding for the servTranslationScheme parameter is specified in section A.4.1.

R8-11 **[599]** Encoding for the primaryTrunkGroupSTS parameter is specified in section A.4.2.

R8-12 **[600]** Encoding for the alternateTrunkGroupSTS parameter is specified in section A.4.3.

R8-13 **[601]** Encoding for the secondAlternateTrunkGroupSTS parameter is specified in section A.4.4.

R8-14 **[602]** Encoding for the overflowRoutingNoSTS parameter is specified in section A.4.5.

8.4.2 Call type

Frequently, the UCS DMS-250 switch categorizes calls as either on- or off-network. Thus, an extension parameter is provided to allow the AIN SCP to indicate to the SSP a new call type to categorize the call. Note that the call type is entered into the billing record, but has no effect on call processing logic.

R8-15 **[132]** The `callType` extension parameter shall contain the calltype to be populated into the CDR. It shall only be returned from the SCP within an `Analyze_Route` message.

R8-16 **[133]** Encoding for the `callType` extension parameter is specified in section A.4.6.

8.4.3 Satellite restriction

In most cases, it is undesirable for long-distance calls to pass through more than one satellite link, as such a link inserts a large, fixed delay into the call. This parameter allows the SCP to indicate that a call should be restricted from routing to an agent which contains a satellite link.

R8-17 **[134]** The `satRestriction` extension parameter shall indicate by its presence that the call in progress shall not be routed to any satellite-based route. It shall only be returned from the SCP within an `Analyze_Route` message.

R8-18 **[135]** Encoding for the `satRestriction` parameter is specified in section A.4.7.

8.4.4 Class of service

Existing feature sets allow the IN/1 SCP to return a class of service (COS) index which is then used for in-switch COS screening, through which some or all route choices may be denied to a subscriber. The class of service parameter allows the AIN SCP to defer COS screening to the SSP.

R8-19 **[136]** The `classOfSvc` extension parameter shall contain an index to the in-switch class of service screening tables. This parameter shall only be returned from the SCP within an `Analyze_Route` message.

R8-20 **[137]** Encoding for the `classOfSvc` extension parameter is specified in section A.4.8.

8.5 Authcode compatibility

The following extension parameter provides compatibility with existing DMS-250 Authcode validation features.

8.5.1 Authcode database index parameter

This parameter transmits the authcode database index (ADIN) value used to select an authorization code database for authcode validation. The ADIN is typically derived from trunk group datafill.

R8-21 **[278v3]** *The* `adin` *extension parameter is supported in the* `Info_Analyzed` *message only if the CAIN protocol stream is* `UCS06` *or higher.*

R8-22 **[279]** Encoding for the `adin` extension parameter is specified in section A.4.14.

8.6 Billing

8.6.1 SCP billing sequence number

This parameter allows the SCP to return to the SSP a 32-bit sequence number which the SSP places into the CDR, allowing downstream matching of SSP billing records with SCP billing records.

R8-23 **[138]** The `billSequenceNumber` extension parameter shall contain a 32-bit binary value which shall be entered into the CDR. This parameter may be transmitted within any response message from the SCP. If transmitted more than once, the last value received at the SSP shall be entered into the CDR.

R8-24 **[139]** Encoding for the `billSequenceNumber` extension parameter is specified in section A.4.9.

8.6.2 CDR Field population

The `amaDigits` extension parameter performs an identical function to the `AMADigitsDialedWC` parameter as described in section 7.7.

R8-25 **[603]** The `amaDigits` extension parameter shall be sent in the `Call_Info_To_Resource` message.

R8-26 **[604]** Encoding for the `amaDigits` extension parameter is specified in section A.4.9.

8.7 Call branding

8.7.1 Call branding parameter

This parameter allows the SCP to indicate to the SSP an announcement that is to be played to the subscriber prior to routing the call to its destination.

R8-27 **[140]** The `callBranding` extension parameter shall contain a single announcement ID. This parameter shall only be transmitted within an `Analyze_Route` message from the SCP.

R8-28 **[141]** The announcement shall not be played more than once.

R8-29 **[142]** The SSP shall detect a non-fatal application error if the tone or announcement is not datafilled on the UCS DMS-250 switch, and shall not play any tone or announcement in this case.

R8-30 **[143]** Encoding for the `callBranding` extension parameter is specified in section A.4.10.

8.8 Universal Access

8.8.1 Universal Access number parameter

This parameter allows the SSP to indicate to the SCP the Universal Access number (typically an 800-type number) dialed by the customer to obtain

switch dial tone. As this number is not the actual address to which the call is placed, it is not transported in the `CollectedAddressInfo` or `CalledPartyID` parameters.

R8-31 **[144v4]** The `universalAccess` extension parameter shall contain the Universal Access number dialled by the customer.

R8-32 **[280]** The format of the parameter depends on the CAIN protocol version. When the version designation of the CAIN protocol version is set to `V1` or higher, the parameter may contain up to 24 digits; if the version designation is `V0`, the previous limit of 10 digits applies.

R8-33 **[145v2]** Encoding for the `universalAccess` extension parameter is specified in section A.4.11.

8.9 PSA SCU support

8.9.1 Connect to SCU parameter

This parameter allows the AIN SCP to indicate to the DMS-250 that it should terminate AIN call processing and enter the Programmable Service Architecture mode by passing control over the call to the PSA Service Control Unit (SCU).

R8-34 **[281]** The `connectToSCU` extension parameter shall cause the SSP to enter PSA call processing. This parameter shall only be transmitted within the `Continue` message from the SCP in response to a query from the `Info_Analyzed` PIC. Once received, the DMS-250 shall discontinue AIN trigger processing and proceed as if the trigger table action had been “query SCU”.

R8-35 **[282]** Encoding for the `connectToSCU` extension parameter is specified in section A.4.12.

8.10 Treatment control

8.10.1 Treatment code parameter

This parameter indicates a treatment set at the SSP prior to triggering the call for AIN, or indicates a treatment to be set as a result of AIN SCP service logic.

R8-36 **[283]** The SSP shall include the `treatment` extension parameter in query messages (one of `Info_Analyzed`, `Info_Collected`, and `O_Feature_Requested`) if a treatment has been set for the call in progress when the call triggers. *The* `treatment` *extension parameter is supported in the* `Info_Analyzed` *message only if the CAIN protocol stream is* `UCS06` *or higher.*

R8-37 **[284]** The SSP shall set the indicated treatment for the call in progress when the `treatment` extension parameter is received from the SCP in either the `Disconnect` message or the `Send_To_Resource` message when used with the `DisconnectFlag` parameter.

R8-38 **[285]** Encoding for the `treatment` parameter is specified in section A.4.15.

8.11 CAIN trigger subscription

8.11.1 CAIN subscription group parameter

This parameter transports the value of the GRPNUM field of table CAINGRP for the group whose trigger subscription caused the call in progress to query.

R8-39 **[286v3]** *The* `cainGroup` *extension parameter is supported in the* `Info_Analyzed` *message only if the CAIN protocol stream is* `UCS06` *or higher.*

R8-40 **[287v3]** When the `cainGroup` extension parameter is received, the trigger subscription for the call in progress will be overwritten by that of the new group. Other group attributes shall not be overwritten. If the call is reoriginated, the new group applies to the new call.

R8-41 **[288]** Encoding for the `cainGroup` extension parameter is specified in section A.4.13.

8.11.2 Call control parameter

This parameter allows the SCP enhanced control over TDP evaluation for the call in progress.

R8-42 **[390]** The SSP shall not perform additional evaluation of the `Info_Analyzed` TDP when the `callCtrl` extension parameter is received in the `Analyze_Route` message with a value of `LEAVE_TDP`.

R8-43 **[391]** The SSP shall not perform additional evaluation of the current TDP when the `callCtrl` extension parameter is received in the `Continue` message with a value of `LEAVE_TDP`.

R8-44 **[392]** The SSP shall not perform additional evaluation of any TDP for the remainder of the call when the `callCtrl` extension parameter is received with a value of `CONT_NOTRIG`. However, TDP evaluation is allowed after reorigination.

R8-45 **[393]** The `callCtrl` extension parameter shall only be transmitted in the `Analyze_Route` and `Continue` messages.

R8-46 **[394]** Encoding for the `callCtrl` extension parameter is specified in section A.4.28.

8.11.3 Subscription Information parameter

The `subscriptionInfo` extension parameter indicates to the SCP the digit type that the SSP triggered on and which subscription method was in use when the query occurred.

R8-47 **[605v2]** *The* `subscriptionInfo` *extension parameter is supported only if the CAIN protocol stream is* UCS08 *or higher.*

R8-48 **[606]** Encoding for the `subscriptionInfo` extension parameter is specified in section A.4.30.

8.12 Outgoing message enhancements

The following extension parameters provide enhanced information from the SSP regarding call state at the point the call in progress sends a message to the SCP.

8.12.1 Collected Address Information parameter

The `collectedAddress` extension parameter performs an identical function to the `CollectedAddressInfo` parameter as described in section 7.23.

R8-49 **[607v2]** *The* `collectedAddress` *extension parameter may be* sent *only if the CAIN protocol stream is* UCS08 *or higher.*

R8-50 **[608]** Encoding for the `collectedAddress` extension parameter is specified in section A.4.31.

8.12.2 Originating trunk information parameter

This parameter transmits the trunk group number, trunk type, and trunk member number for the agent on which the call originated.

R8-51 **[289v2]** *The* `origTrunkInfo` *extension parameter is supported in the* `Info_Analyzed` *message only if the CAIN protocol stream is* `UCS06` *or higher.*

R8-52 **[290]** Encoding for the `origTrunkInfo` extension parameter is specified in section A.4.18.

8.12.3 Terminating trunk information parameter

This parameter transmits information regarding the terminating trunk (trunk group number, trunk type, and trunk member).

R8-53 **[291v2]** (Deleted)

R8-54 **[292]** Encoding for the `termTrunkInfo` extension parameter is specified in section A.4.18.

8.12.4 Busy route parameter

This parameter identifies the route (or routing number) which caused the `Network_Busy`, `O_Called_Party_Busy`, or `O_No_Answer` detection points to apply to the call in progress. The number of (failed) attempts to complete the call to that route (or routing number) may also be specified.

R8-55 **[293]** The number of tries indicator shall be the number of repeat queries the SSP generates as a result of failing to route the call (including busy/no-answer scenarios).

R8-56 **[294]** The route index shall be included if the routing attempt was to a route index returned in a trunk group parameter (for example, `PrimaryTrunkGroup`).

R8-57 **[295v2]** The routing number field of the extension parameter shall indicate the number outpulsed on the terminating trunk, if available, which must have been an outpulse number.

R8-58 **[296]** The `busyRoute` parameter shall only be transmitted within the `Network_Busy`, `O_Called_Party_Busy`, and `O_No_Answer` request messages sent from the SSP.

R8-59 **[297]** Encoding for the `busyRoute` extension parameter is specified in section A.4.19.

8.12.5 Reoriginated call indicator parameter

This parameter is transmitted in an SSP query message to indicate that the call in progress is the result of a reorigination of a previous call.

R8-60 **[298v3]** *The* `reorigCall` *extension parameter is supported in the* `Info_Analyzed` *message only if the stream designation of the CAIN protocol version is* `UCS06` *or higher.*

R8-61 **[299]** Encoding for the `reorigCall` extension parameter is specified in section A.4.16.

8.12.6 Number of Reoriginations parameter

This parameter is sent to the SCP to indicate the number of times a user has reoriginated.

R8-62 **[609]** The `numReorig` extension parameter is sent in the `O_Mid_Call` message.

R8-63 **[610]** Encoding for the `numReorig` extension parameter is specified in section A.4.32.

8.12.7 T1 timeout overflow requery indicator parameter

This parameter is transmitted in an SSP T1 timeout requery message to indicate that the current query is an overflow requery message resulting from a T1 timeout detected for the original query message.

R8-64 **[395]** The `t1Overflow` extension parameter shall indicate by its presence that a T1 timeout was detected for the original query message.

R8-65 **[396]** The `t1Overflow` extension parameter shall only be transmitted in SSP T1 timeout requery messages.

R8-66 **[397]** Encoding for the `t1Overflow` extension parameter is specified in section A.4.26.

8.12.8 Local Number Portability (LNP) information received indicator parameter

This parameter indicates to the SCP that a Local Number Portability (LNP) check has been performed at a preceding switch prior to the current query.

R8-67 **[398]** The `lnpReceived` extension parameter shall indicate by its presence that incoming LNP information has been received for the call in progress.

R8-68 **[399v2]** *This extension parameter is supported in UCS07 and beyond only if the stream designation of the CAIN protocol version is* `UCS07` *or higher.*

R8-69 **[400]** Encoding for the `lnpReceived` extension parameter is specified in section A.4.27.

8.12.9 Jurisdiction Information parameter

The `jurisdictionInformation` extension parameter performs an identical function to the `JurisdictionInformation` parameter as described in section 7.37.

R8-70 **[611v2]** The `jurisdictionInformation` extension parameter may be sent in the `Origination_Attempt`, `O_Feature_Requested`, `Info_Collected`, `Network_Busy`, `O_Called_Party_Busy`, `O_No_Answer`, and `O_Mid_Call` TDP-R messages *only if the CAIN protocol stream is* UCS08 *or higher.*

R8-71 **[612]** Encoding for the `jurisdictionInformation` extension parameter is specified in section A.4.33.

8.12.10 Switch Identifier parameter

This parameter contains the switchID that exists for the call.

R8-72 **[731]** The `switchID` extension parameter may be sent in all TDP-R messages *only if the CAIN protocol stream is* UCS0*9 or higher.*

R8-73 **[732]** Deleted.

R8-74 **[733]** Deleted.

R8-75 **[734]** Encoding for the `switchID` extension parameter is specified in section A.4.34.

8.12.11 Billing Number parameter

This parameter contains the billing number that exists for the call.

R8-76 **[735]** The `billingNumber` extension parameter may be sent in all TDP-R messages *only if the CAIN protocol stream is* UCS0*9 or higher.*

R8-77 **[736]** The `billingNumber` extension parameter may be received in the `Analyze_Route` message.

R8-78 **[737]** The digits used to populate this extension parameter may have been collected by the Virtual IP (VIP).

R8-79 **[738]** The `billingNumber` extension parameter may be sent with a Nature of Address of: CARD, AUTH, ACCT, PIN, N00; and Numbering Plan of PRVT.

R8-80 **[739]** The `billingNumber` extension parameter may be received with a Nature of Address of: CARD, AUTH, ACCT, PIN, N00, ANI; and Numbering Plan of PRVT.

R8-81 **[740]** Encoding for the `billingNumber` extension parameter is specified in section A.4.35.

8.12.12 Account Code parameter

This parameter contains the account code that exists for the call.

R8-82 **[741]** The `accountCode` extension parameter may be sent in the `O_Feature_Requested, Information_Collected, Information_Analyzed` and `O_Mid_Call` messages *only if the stream designation of the CAIN protocol version is* UCS0*9 or higher.*

R8-83 **[742]** The `accountCode` extension parameter may be received in the `Analyze_Route` message.

R8-84 **[743]** The digits used to populate this extension parameter may have been collected by the Virtual IP (VIP).

R8-85 **[744]** The `accountCode` extension parameter shall have a Nature of Address of UNKNOWN and Numbering Plan of ISDN.

R8-86 **[745]** Encoding for the `accountCode` extension parameter is specified in section A.4.36.

8.12.13 PIN Digits parameter

This parameter contains the PIN digits that exist for the call.

R8-87 **[746]** The `pinDigits` extension parameter may be sent in the `O_Feature_Requested, Information_Collected` and `Information_Analyzed` messages *only if the stream designation of the CAIN protocol version is* UCS0*9 or higher.*

R8-88 **[747]** The `pinDigits` extension parameter may be received in the `Analyze_Route` message.

R8-89 **[748]** The digits used to populate this extension parameter may have been collected by the Virtual IP (VIP).

R8-90 **[749]** The `pinDigits` extension parameter shall have a Nature of Address of UNKNOWN and Numbering Plan of ISDN.

R8-91 **[750]** Encoding for the `pinDigits` extension parameter is specified in section A.4.37.

8.12.14 Connected Time indicator parameter

This parameter indicates to the SCP how much time has elapsed since the call was answered, in minutes, seconds, and tenths of seconds. This provides the same functionality as the `ConnectTime` standard parameter as discussed in section 6.49 of GR-1299-CORE.

R8-92 **[613]** The `connectTime` extension parameter may be sent in the `Timeout` and `O_Disconnect` messages.

R8-93 **[614]** Encoding for the `connectTime` extension parameter is specified in section A.4.38.

8.13 Response message enhancements

The following parameter provides enhanced information in response messages sent from the SCP.

8.13.1 Reorigination allowed indicator parameter

This parameter is sent to the SSP to indicate whether reorigination is allowed for the call in progress by identifying the number of successive reoriginations allowed. Note that call reorigination is still subject to in-switch feature and datafill restrictions which may override this parameter.

R8-94 **[300]** The `reorigAllowed` extension parameter contains an integer denoting the number of successive reoriginations allowed, with 0 indicating that reorigination is prohibited. This parameter shall only be transmitted within an `Analyze_Route` response message from the SCP.

R8-95 **[301]** Encoding for the `reorigAllowed` extension parameter is specified in section A.4.17.

8.14 Event Detection Point (EDP) control

The following extension parameters allow the SCP to instruct the SSP as to what actions are to be performed when the `Network_Busy`, `O_Called_Party_Busy`, `O_No_Answer` and `O_Mid_Call` EDPs are encountered.

8.14.1 Network_Busy EDP actions parameter

The `networkBusyActions` parameter specifies the action to take based on the routing criteria for the `Network_Busy` EDP.

R8-96 **[401]** The route available (RTEAVAIL) field shall indicate the action to perform if additional routes are available when the `Network_Busy` EDP is encountered.

R8-97 **[402]** The routes done (RTESDONE) field shall indicate the action to perform if no additional routes are available.

R8-98 **[403v2]** The terminating route generalized no circuit (TERMRTE_GNCT) field shall indicate the action to take when the terminating route choice is busy. Direct termination routing must have been used to obtain this route.

R8-99 **[404]** The `networkBusyActions` extension parameter shall only be transmitted within the `Request_Report_BCM_Event` message from the SCP.

R8-100 **[405]** Encoding for the `networkBusyActions` extension parameter is specified in section A.4.20.

8.14.2 O_Called_Party_Busy EDP actions parameter

The `oCalledPartyBusyActions` parameter specifies the action to take based on the routing criteria for the `O_Called_Party_Busy` EDP.

R8-101 **[406]** The route available (RTEAVAIL) field shall indicate the action to perform if additional routes are available when the `O_Called_Party_Busy` EDP is encountered.

R8-102 **[407]** The routes done (RTESDONE) field shall indicate the action to perform if no additional routes are available when the `O_Called_Party_Busy` EDP is encountered.

R8-103 **[408]** The `oCalledPartyBusyActions` extension parameter shall only be transmitted within the `Request_Report_BCM_Event` message from the SCP.

R8-104 **[409]** Encoding for the `oCalledPartyBusyActions` extension parameter is specified in section A.4.21.

8.14.3 O_No_Answer EDP actions parameter

The `oNoAnswerActions` parameter specifies the action to take based on the routing criteria for the `O_No_Answer` EDP.

R8-105 **[410]** The route available (RTEAVAIL) field shall indicate the action to perform if additional routes are available when the `O_No_Answer` EDP is encountered.

R8-106 **[411]** The routes done (RTESDONE) field shall indicate the action to perform if no additional routes are available when the `O_No_Answer` EDP is encountered.

R8-107 **[412]** The `oNoAnswerActions` extension parameter shall only be transmitted within the `Request_Report_BCM_Event` message from the SCP.

R8-108 [413] Encoding for the `oNoAnswerActions` extension parameter is specified in section A.4.22.

8.14.4 EDP digit buffer indicator parameter

The `edpBuffer` parameter indicates that digits shall be buffered when an EDP-Request message is sent to the SCP.

R8-109 [414] The `edpBuffer` extension parameter shall indicate by its presence that digits should be buffered for possible conversation when an EDP-R is sent. It shall only be transmitted in the `Request_Report_BCM_Event` message from the SCP.

R8-110 [415] Digit buffering is not supported for the `O_No_Answer` EDP.

R8-111 [416] Encoding for the `edpBuffer` extension parameter is specified in section A.4.23.

8.14.5 Switch_Hook_Flash Enabled Legs parameter

The `shfelegs` parameter specifies which legs may be monitored for the SwitchHookFlash event.

R8-112 [844] The `shfelegs` extension parameter may be sent by the SCP to the SSP to indicate which legs of a call to monitor for the SwitchHookFlash event.

R8-113 [845] The `shfelegs` extension parameter is subject to the SSP arming restrictions for the SwitchHookFlash event and may be ignored in some Call Configurations and/or for certain legs. If all specified legs are ignored then the SwitchHookFlash event will not be armed.

R8-114 [846] Encoding for the `shfelegs` extension parameter is specified in section A.4.43.

8.15 Multi-switch Business Group support

8.15.1 MBG/NETINFO parameter

When Multi-switch Business Group (MBG) calls are received from an SL-100 switch, the business group information is received in the optional NETINFO parameter of the incoming ISUP Initial Address Message (IAM).

R8-115 [417] The `netinfo` extension parameter shall transmit to the SCP the business group information for the call received from the NETINFO parameter of the incoming IAM. This extension parameter shall be transmitted by the SSP only in the `Info_Collected`, `O_Feature_Requested`, `Info_Analyzed`, `Network_Busy`, `O_Called_Party_Busy`, and `O_No_Answer` messages.

R8-116 [418] The SCP may return the `netinfo` extension parameter with new business group information to be sent in an outgoing ISUP IAM. This extension parameter shall be transmitted by the SCP only in the `Analyze_Route` message.

R8-117 [419] Encoding for the `netinfo` extension parameter is specified in section A.4.25.

8.16 Global Inter Machine Trunk (IMT) support

8.16.1 Universal Translations index parameter

Global Inter Machine Trunks (IMTs) use Universal Translations for routing. This extension parameter indicates the system of Universal Translations is or will be in use for the call in progress.

R8-118 [420v2] The `univIdx` extension parameter shall indicate to the SCP the Universal Translation scheme currently in use at the SSP.

R8-119 [421] The `univIdx` extension parameter shall indicate to the SSP which Universal Translations system to use. This parameter shall be transmitted by the SCP only in the `Analyze_Route` and `Send_To_Resource` messages.

R8-120 [422] Encoding for the `univIdx` extension parameter is specified in section A.4.24.

8.17 Intelligent Peripheral support

8.17.1 Send_To_Resource Connection Type parameter

This parameter indicates what type of Intelligent Peripheral Interface (IPI) connection protocol is to be used to establish communication between the SCP, SSP, and an IP resource.

R8-121 [615] The `strConnectionType` extension parameter may be sent in the `Send_To_Resource` and `Connect_To_Resource` messages.

R8-122 [616] Encoding for the `strConnectionType` extension parameter is specified in section A.4.39.

8.18 Virtual IP support

8.18.1 Pretranslator Name parameter

This parameter indicates the new pretranslator to be used for the call in progress. This pretranslator is used for the remained of the call including reoriginations. All ADDR collectibles obtained through VIP processing are subjected to the indicated pretranslator.

R8-123 [751] The `pretranslatorName` may be sent in the `Send_To_Resource` and `Connect_To_Resource` messages.

R8-124 [752] Encoding for the `pretranslatorName` extension parameter is specified in section A.4.40.

8.19 CAIN Pretranslator support

8.19.1 CAIN Pretranslator error scenario parameter

This parameter indicates to the SCP that the SSP was performing CAIN pretranslations and an error occurred. Any collected digits are sent.

R8-125 [617] The `cainPRT` extension parameter is sent in the `O_Feature_Requested` message *only if the stream designation of the CAIN protocol version is set to* UCS08 *or higher.*

R8-126 [618] Encoding of the `cainPRT` extension parameter is specified in section A.4.41.

8.20 Code Gapping Support

8.20.1 Automatic Code Gapping (ACG) Re-query parameter

The `acgRequery` parameter, when present, indicates to the SCP that the current query is an re-query message resulting from the original query being blocked by an ACG control.

R8-127 [753] The `acgRequery` extension parameter shall indicate by its presence that the present query is a re-query resulting from the original query being blocked by an ACG.

R8-128 [754] The `acgRequery` extension parameter may be sent in all TDP-R messages.

R8-129 [755] Encoding for the `acgRequery` extension parameter is specified in section A.4.42.

8.21 Access Type support

8.21.1 Access Type parameter

The `accessType` extension parameter provides type of access information.

R8-130 [867] The `accessType` extension parameter is sent *only if all of the following three conditions are true:*

- the SEND_SPECIAL_AA parameter in table CAINPARM is set to Y.
- option LDCUST is present on the originating trunk group (available on an Equal Access Network Trunk (EANT) only).
- the call has triggered at OFFHKDEL.

R8-131 [868] The `accessType` extension parameter belongs to the `infoCollectedExtensionAa` family.

R8-132 [869] The `accessType` extension parameter shall be sent to the SCP only in an `Info_Collected` message.

R8-133 **[870]** Encoding for the `accessType` extension parameter is specified in section A.5.1.

Chapter 9: Messaging scenarios

9.1 Introduction

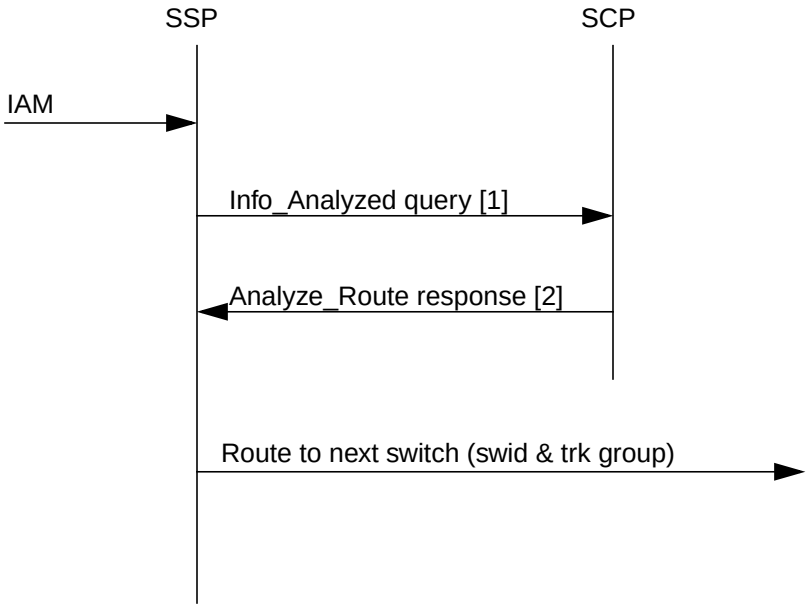
This chapter offers examples of service switching point (SSP) to service control point (SCP) messaging scenarios which may be expected to take place. It is not exhaustive, and applies only to those messages supported in the current release. It is intended to clarify the SSP-SCP interactions and the operation of the call model.

In each of the following sections, the illustrations attempt to convey both the sequence of messages exchanged, including specific parameters of interest, and the present state of the call in the SSP upon sending or receiving a message.

9.2 FGD call with normal routing

In Figure 1, an inbound SS7 FGD call triggers to provide a custom dial plan with translation at the SCP. The SCP computes a route for the call to continue through the network.

Figure 1 FGD call with direct-termination routing



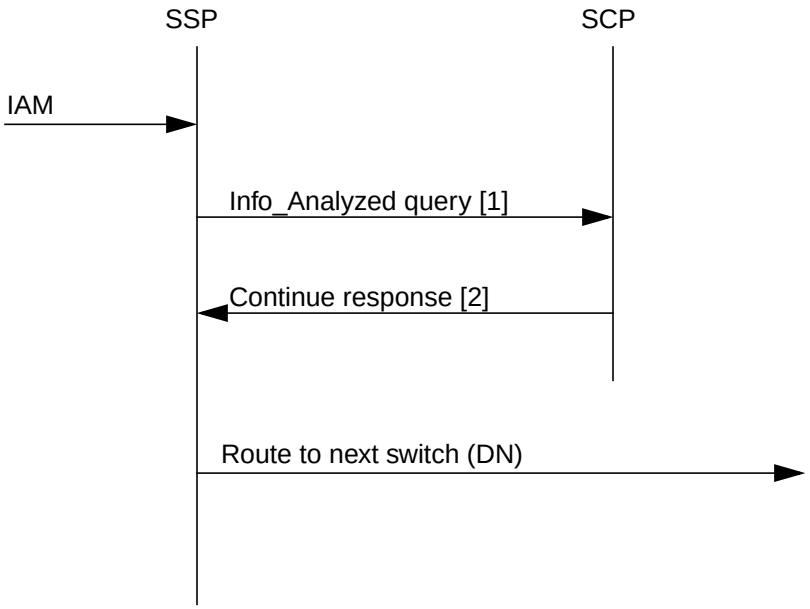
1. The following information is passed in the query message:
   - UserID (CLLI trunk group number of the FGD trunk)
   - BearerCapability (bearer capability of the FGD trunk)
   - CalledPartyID (called address)
   - TriggerCriteriaType (customized dial plan)
   - ChargeNumber (ANI)
   - CallingPartyID (ANI)
2. The SCP provides the following information in return:
   - PrimaryTrunkGroup (switch ID/trunk group)

   In addition, alternate routes might be provided, as well as replacement information (for example, a different charge number than that provided by the SSP).

9.3 FGD call with default routing

In Figure 2, the SS7 FGD call triggers for ANI screening at the SCP. The SCP allows the call to continue using in-switch routing.

Figure 2 FGD call with default routing

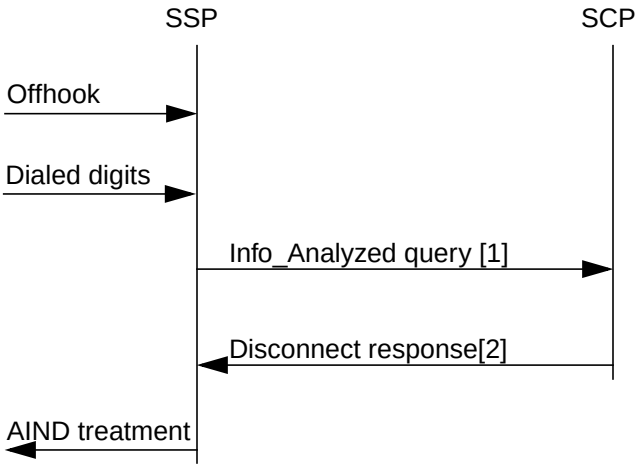


1. The following information is passed in the query message:
   - UserID (CLLI trunk group number of the FGD trunk)
   - BearerCapability (bearer capability of the FGD trunk)
   - CalledPartyID (called address)
   - TriggerCriteriaType (customized dial plan)
   - ChargeNumber (ANI)
   - CallingPartyID (ANI)
   - CollectedAddressInfo (dialed digits)
2. In return, the SCP sends a Continue message with no parameters. The SSP then performs in-switch translation and routing of the called party's number.

9.4 DAL N00 call, disconnected

In Figure 3, an N00 call originated on a DAL trunk triggers for N00 number translation and is treated when the SCP determines the N00 number is not valid.

Figure 3 DAL N00 call with disconnect

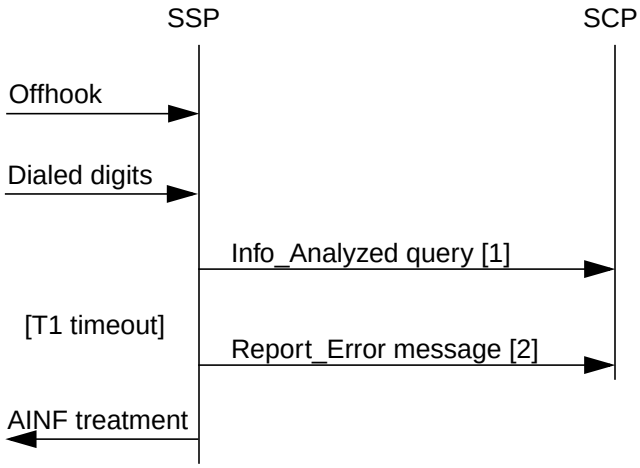


1. The following information is passed in the query message:
   - UserID (CLLI trunk group number of the DAL trunk)
   - BearerCapability (bearer capability of the DAL trunk)
   - CalledPartyID (called address)
   - TriggerCriteriaType (sds_n00)
   - ChargeNumber (dialed/filed authcode)
   - CallingPartyID (SNPA of the DAL trunk)
   - CollectedAddressInfo (dialed digits)
2. By returning the Disconnect message without the treatment extension parameter, the SCP causes the SSP to provide final treatment of the call using the AIN Disconnect treatment (AIND). If the treatment extension parameter is included, then the specified treatment is applied to the call. As an alternative, the Send_to_Resource message could be used to return a specific tone or announcement to be played, after which the SSP would take down the call.

9.5 DAL N00 call, SCP failure

In Figure 4, the SCP fails to return a response to the SSP's query prior to the T1 timer expiring. As a result, the SSP sends an error message to the SCP and takes down the call after applying a treatment.

Figure 4 DAL N00 call, SCP failure

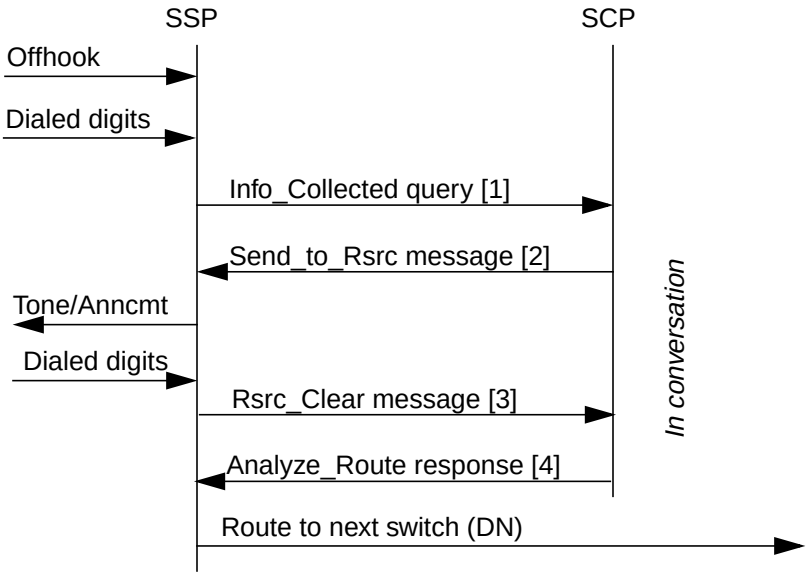


1. The following information is passed in the query message:
   - UserID (CLLI trunk group number of the DAL trunk)
   - BearerCapability (bearer capability of the DAL trunk)
   - CalledPartyID (called address)
   - TriggerCriteriaType (sds_n00)
   - ChargeNumber (dialed/filed authcode)
   - CallingPartyID (SNPA of the DAL trunk)
   - CollectedAddressInfo (dialed digits)
2. Following expiration of the T1 timer, the SSP sends the Report_Error message with an ApplicationErrorString parameter indicating a cause of "response message timer expired". This message is transmitted in a unidirectional TCAP package because a transaction was never established between the SSP and SCP. The SSP then treats the call with AIN Final treatment (AINF).

9.6 DAL VPN call with conversation

In Figure 5, the SCP enters conversation upon receiving the SSP's query in order to collect additional digits. As a result, the SSP collects and reports the digits to the SSP in order to elicit final routing information.

Figure 5 DAL VPN call with conversation

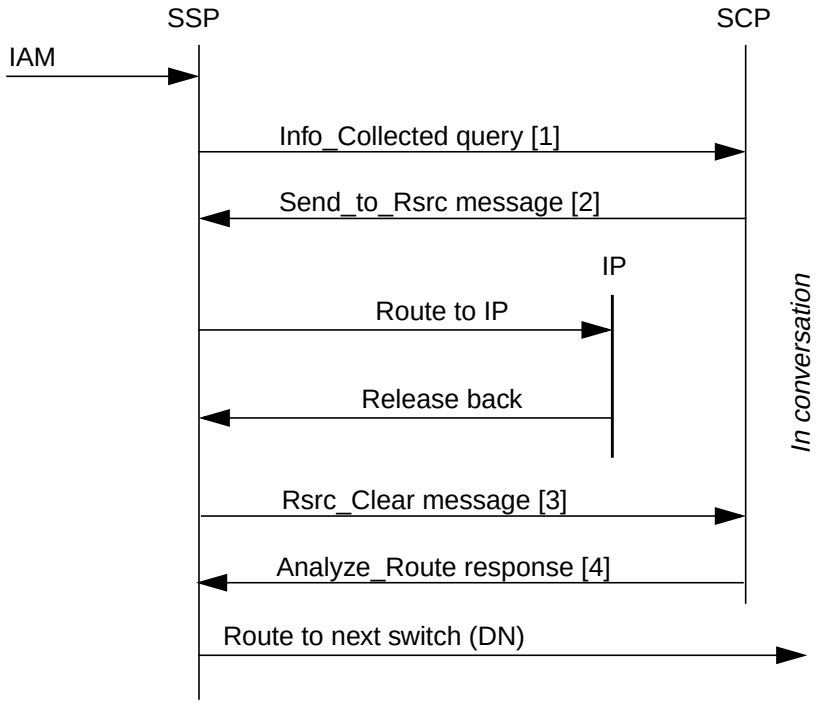


1. The following information is passed in the query message:
   - UserID (CLLI trunk group number of the DAL trunk)
   - BearerCapability (bearer capability of the DAL trunk)
   - ChargeNumber (dialed/filed authcode)
   - TriggerCriteriaType (sio_addr)
   - CallingPartyID (SNPA of the DAL trunk)
   - CollectedAddressInfo (dialed digits)
2. The SCP sends a Send_to_Resource message in conversation to collect a PIN code. The SSP plays the indicated tone/announcement and collects digits.
3. The SSP then returns the collected digits to the SCP in a conversational Resource_Clear message.
4. The SCP responds with an Analyze_Route response containing the destination DN.

9.7 FGD call using the IP in conversation

In Figure 6, the SCP sends the caller to an IP for extended interaction, while remaining in conversation with the SSP. Only once the IP releases the call does the SSP return control to the SCP for final routing instructions.

Figure 6 FGD call using the IP in conversation

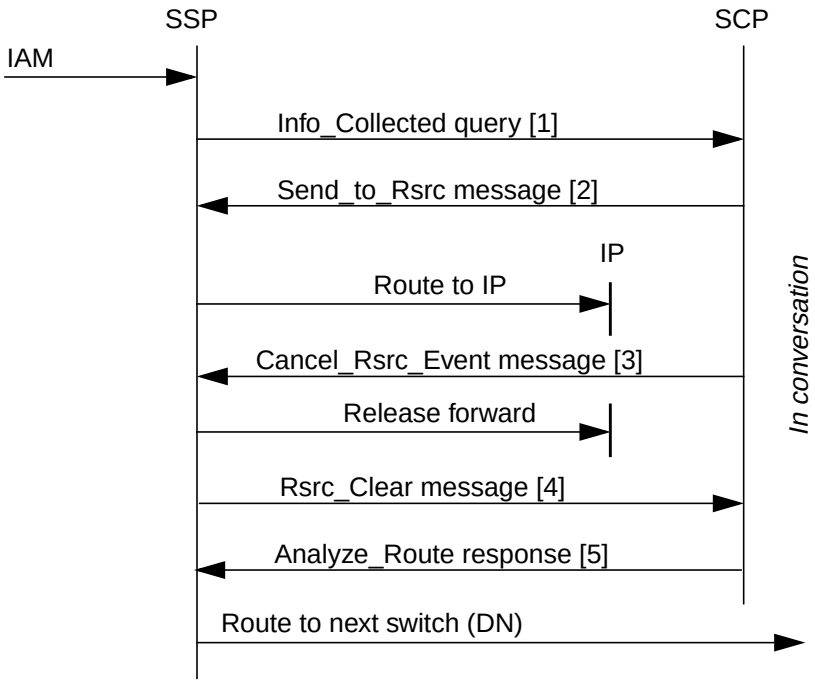


1. The following information is passed in the query message:
   - UserID (CLLI trunk group number of the DAL trunk)
   - BearerCapability (bearer capability of the DAL trunk)
   - ChargeNumber (dialed/filed authcode)
   - TriggerCriteriaType (sio_addr)
   - CallingPartyID (SNPA of the DAL trunk)
   - CollectedAddressInfo (dialed digits)
2. The SCP sends a Send_to_Resource message in conversation to collect a PIN code. The SSP extends a call leg to the IP.
3. When the IP interaction terminates (the IP releases back to the SSP), the SSP returns control to the SCP in a conversational Resource_Clear message.
4. The SCP responds with an Analyze_Route response containing the destination DN.

9.8 FGD call with IP interaction cancelled

In Figure 7, the SCP sends the caller to an IP for extended interaction, while remaining in conversation with the SSP. The SCP then cancels the IP interaction before the IP has released the call back, causing the SSP to terminate the IP leg.

Figure 7 FGD call with IP interaction canceled

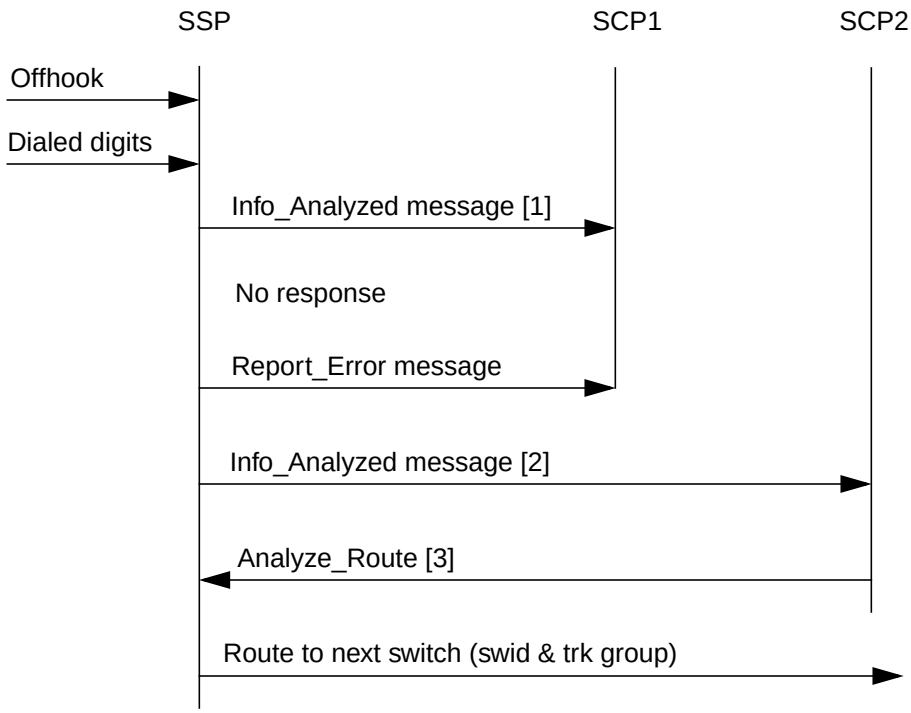


1. The following information is passed in the query message:
   - UserID (CLLI trunk group number of the DAL trunk)
   - BearerCapability (bearer capability of the DAL trunk)
   - ChargeNumber (dialed/filed authcode)
   - TriggerCriteriaType (sio_addr)
   - CallingPartyID (SNPA of the DAL trunk)
   - CollectedAddressInfo (dialed digits)
2. The SCP sends a Send_to_Resource message in conversation to collect a PIN code. The SSP extends a call leg to the IP.
3. The SCP then sends a Cancel_Resource_Event message to end the IP interaction prematurely.
4. The SSP terminates the IP leg and returns control to the SCP in a conversational Resource_Clear message.
5. The SCP responds with an Analyze_Route response containing the destination DN.

9.9 DAL N00 call with T1 timeout overflow requery

In Figure 8, an N00 call originated on a DAL trunk triggers for N00 number translation. A T1 timeout occurs, and the SSP sends a T1 timeout re-query to a second SCP.

Figure 8 DAL N00 call with T1 timeout overflow requery

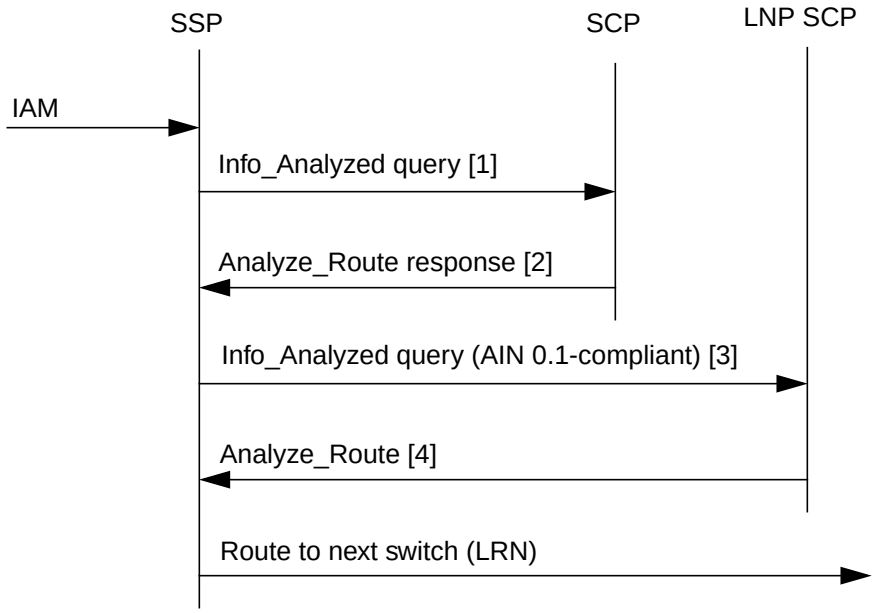


1. The following information is passed to SCP1 in the query message:
   - UserID (CLLI trunk group number of the DAL trunk)
   - BearerCapability (bearer capability of the DAL trunk)
   - CalledPartyID (called address)
   - TriggerCriteriaType (sds_n00)
   - ChargeNumber (dialed/filed authcode)
   - CallingPartyID (SNPA of the DAL trunk)
   - CollectedAddressInfo (dialed digits)
2. A T1 timeout is detected by the SSP when SCP1 fails to respond. The SSP sends a Report_Error to notify SCP1 that an error occurred. The SSP sends the same query information, along with the t1Overflow extension parameter, to SCP2.
3. The SCP provides the following information in return:
   - PrimaryTrunkGroup (switch ID/trunk group)

9.10 FGD call, Info_Analyzed TDP reevaluation, Office_Code trigger

In Figure 9, an inbound SS7 FGD call triggers to provide a custom dial plan with translation at the SCP. The SCP returns an address for routing. The SSP reevaluates the `Info_Analyzed` TDP and triggers at the `Office_Code` trigger.

Figure 9 FGD call, Info_Analyzed TDP reevaluation, Office_Code trigger

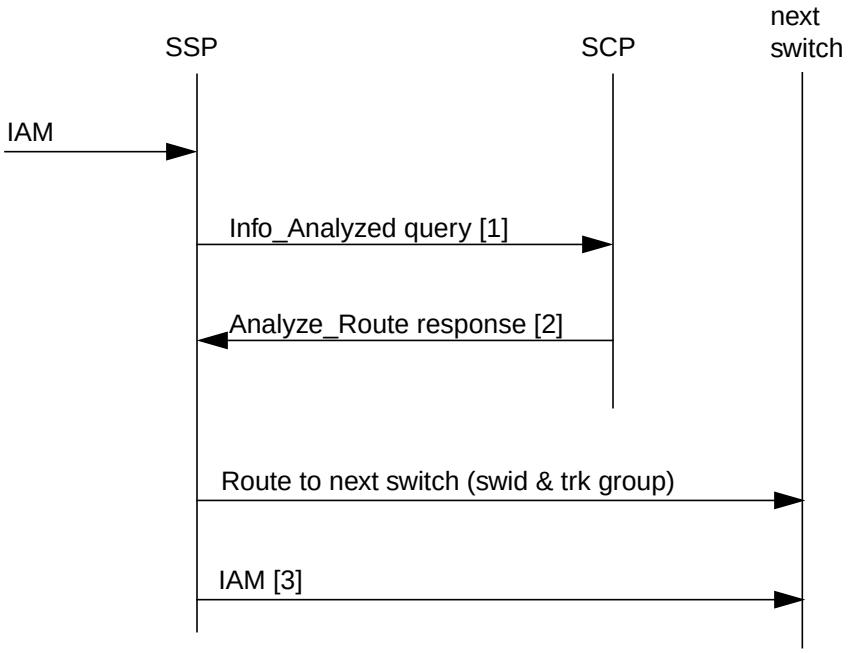


1. The following information is passed in the query message:
   - UserID (CLLI trunk group number of the FGD trunk)
   - BearerCapability (bearer capability of the FGD trunk)
   - CalledPartyID (called address)
   - TriggerCriteriaType (customized dial plan)
   - ChargeNumber (ANI)
   - CallingPartyID (ANI)
2. The SCP provides the following information in return:
   - CalledPartyID (new called address)
3. Reevaluation of the Info_Analyzed TDP triggers at the Office_Code trigger. An AIN 0.1-compliant Info_Analyzed message is sent to the LNP SCP
4. The LNP SCP responds with the following:
   - CalledPartyID (Location Routing Number [LRN])

9.11 FGD call with LNP information and direct-termination routing

In Figure 10, an inbound SS7 FGD call triggers to provide a custom dial plan with translation at the SCP. The SCP computes a route for the call to continue through the network. The SCP also performs an LNP check and determines that the called address has been ported.

Figure 10 FGD call with LNP information and direct-termination routing

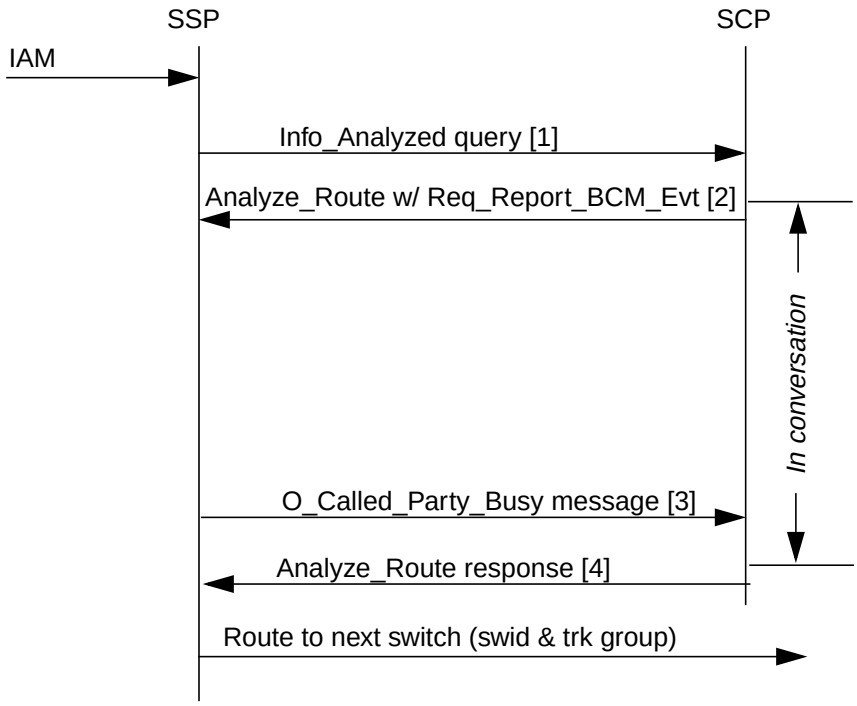


1. The following information is passed in the query message:
   - UserID (CLLI trunk group number of the FGD trunk)
   - BearerCapability (bearer capability of the FGD trunk)
   - CalledPartyID (called address)
   - TriggerCriteriaType (customized dial plan)
   - ChargeNumber (ANI)
   - CallingPartyID (ANI)
2. The SCP provides the following information in return:
   - PrimaryTrunkGroup (switch ID/trunk group)
   - ForwardCallIndicator (bit M set)
   - CalledPartyID (LRN)
   - GenericAddress (called address)
3. The IAM to the next switch contains the following information:
   - CalledPartyID (LRN)
   - ForwardCallIndicator (bit M set)
   - GenericAddress (called address)

9.12 FGD call with EDP-Request

In Figure 11, the SCP instructs the SSP to arm EDPs. When an armed EDP is reached, the SSP sends an EDP-Request and the SCP responds.

Figure 11 FGD call with EDP-Request

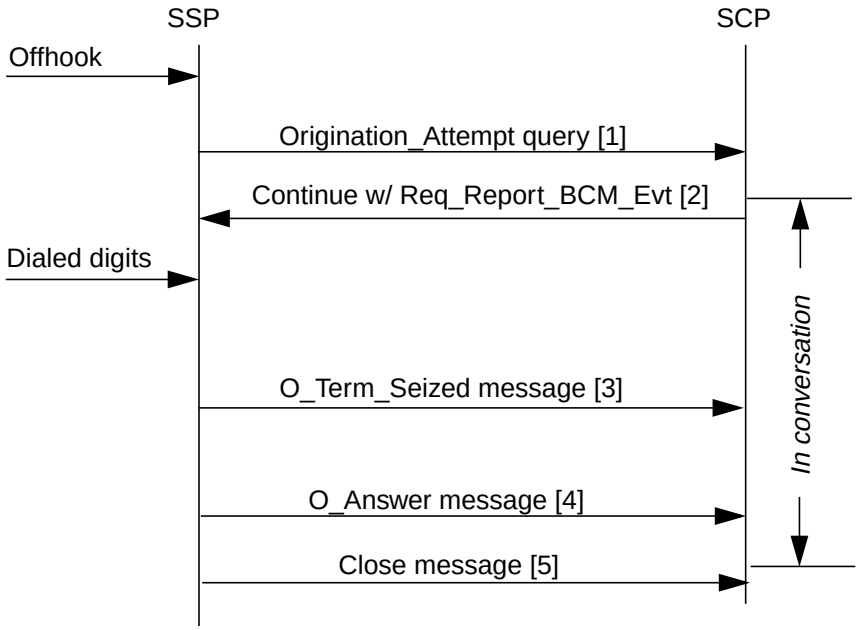


1. The SSP sends an Info_Analyzed TDP-Request to the SCP.
2. The SCP sends the SSP a conversation package containing an Analyze_Route component followed by a Request_Report_BCM_Event component which arms the O_Called_Party_Busy EDP.
3. The SSP attempts to send the originator to the route specified in the Analyze_Route component, but an O_Called_Party_Busy condition is detected. The SSP sends an O_Called_Party_Busy EDP-Request in a conversation package and deactivates EDPs.
4. The SCP provides the following information in a response package containing an Analyze_Route component only:
   - PrimaryTrunkGroup (switch ID/trunk group)

9.13 DAL call with EDP-Notifications

In Figure 12, the SCP instructs the SSP to arm EDPs. When an armed EDP is reached, the SSP sends an EDP-Notification. When all active EDPs are passed, the SSP sends a Close message to the SCP.

Figure 12 DAL call with EDP-Notifications

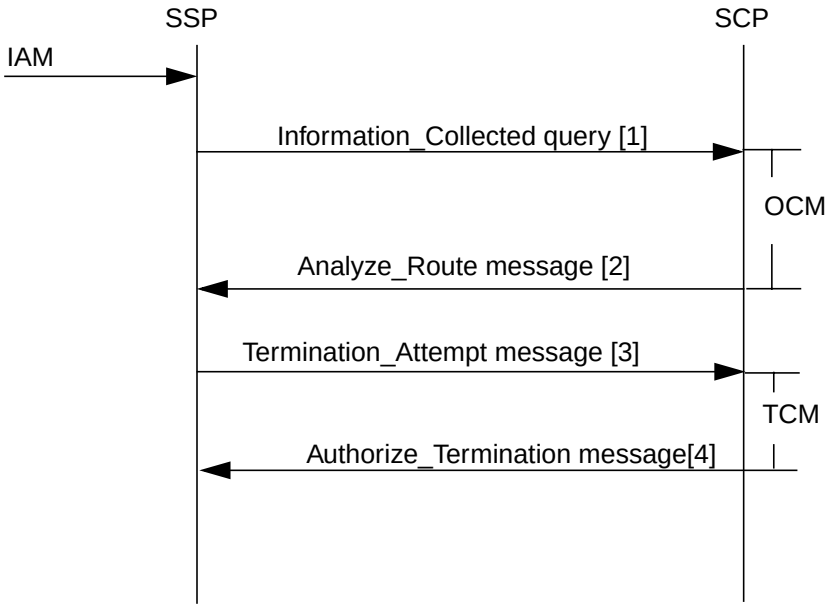


1. The SSP sends an Origination_Attempt TDP-Request to the SCP.
2. The SCP sends the SSP a conversation package containing a Continue component followed by a Request_Report_BCM_Event component which arms the O_Term_Seized and O_Answer EDPs.
3. After the originator dials an address and the terminator is seized, the SSP sends an O_Term_Seized EDP-Notification to the SCP in a conversation package. EDPs are not deactivated.
4. After the terminator answers, the SSP sends an O_Answer EDP-Notification to the SCP in a conversation package. EDPs are not deactivated.
5. Since no more EDPs are reachable after O_Answer, the SSP sends a Close message to the SCP in a response package and deactivates EDPs.

9.14 SS7 IMT with an OCM completed transaction followed by a TCM transaction

In Figure 13, the SCP instructs the SSP to route the call. When the SSP selects the terminating trunk group, the SSP sends the Termination_Attempt query to the SCP. The SCP allows the call to terminate.

Figure 13 SS7 IMT with an OCM completed transaction followed by a TCM transaction

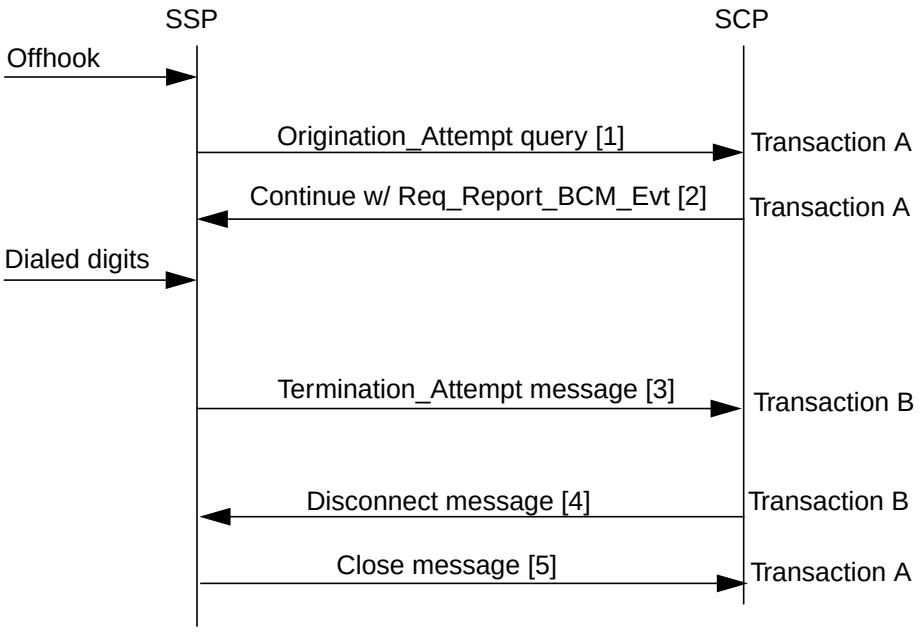


1. The SSP sends an Information_Collected TDP-Request to the SCP.
2. The SCP sends the SSP a response package containing an Analyze_Route component.
3. The SSP sends a Termination_Attempt TDP-Request to the SCP.
4. The SCP sends the SSP a response package containing an Authorize_Termination component.

9.15 DAL call with EDP-Notifications and TCM

In Figure 14, the SCP instructs the SSP to arm EDPs. When the SSP selects the terminating trunk group, the SSP sends the Termination_Attempt message to the SCP. The SCP does not allow the call to progress and the SSP closes the EDP transaction.

Figure 14 DAL call with EDP-Notifications and TCM

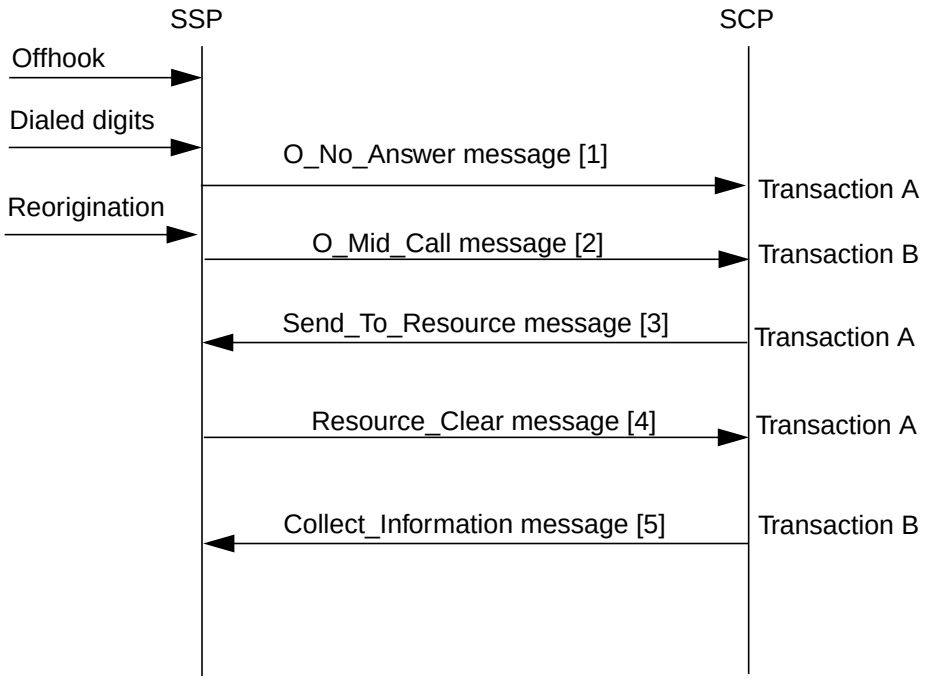


1. The SSP sends an Origination_Attempt TDP-Request to the SCP.
2. The SCP sends the SSP a conversation package containing a Continue component followed by a Request_Report_BCM_Event component which arms the O_Term_Seized and O_Answer EDPs.
3. After the originator dials an address and the terminating trunk group is selected. The Termination_Attempt TDP-Request is sent to the SCP.
4. The SCP responds with a Disconnect message.
5. Since the SCP directs the SSP to disconnect the call, no more EDPs are reachable and the SSP sends a Close message to the SCP and it deactivates EDPs.

9.16 DAL call with O_Mid_Call and Collect_Information

In Figure 15, the SCP instructs the SSP use the pre-query dialing plan after reorigination is complete.

Figure 15 DAL call with O_Mid_Call and Collect_Information

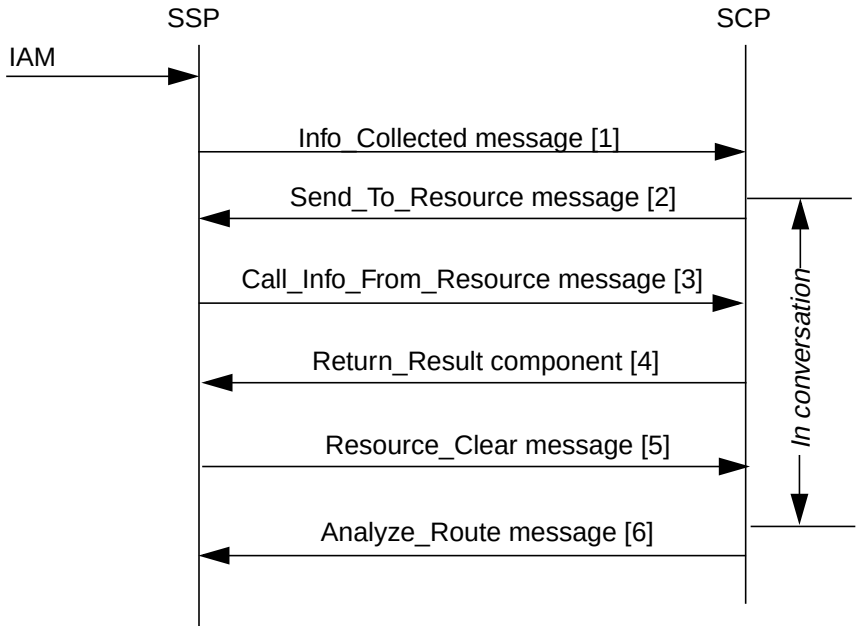


1. The SSP sends an O_No_Answer TDP-Request to the SCP.
2. The SSP sends an O_Mid_Call TDP-Request to the SCP
3. The SCP sends a conversation package with a Send_To_Resource operation.
4. The SSP responds to [4] with a Resource_Clear in a response package with a ClearCause of userAbandon.
5. The SCP directs the SSP to instantiate the dialing plan in-force with a Collect_Information message.

9.17 FGD call with IP Interactions with Call_Info_From_Resource

In Figure 16, the SCP instructs the SSP to connect to an IP. The IP communicates intermediate information to the SCP through a Call_Info_From_Resource.

Figure 16 FGD call with IP Interactions with Call_Info_From_Resource

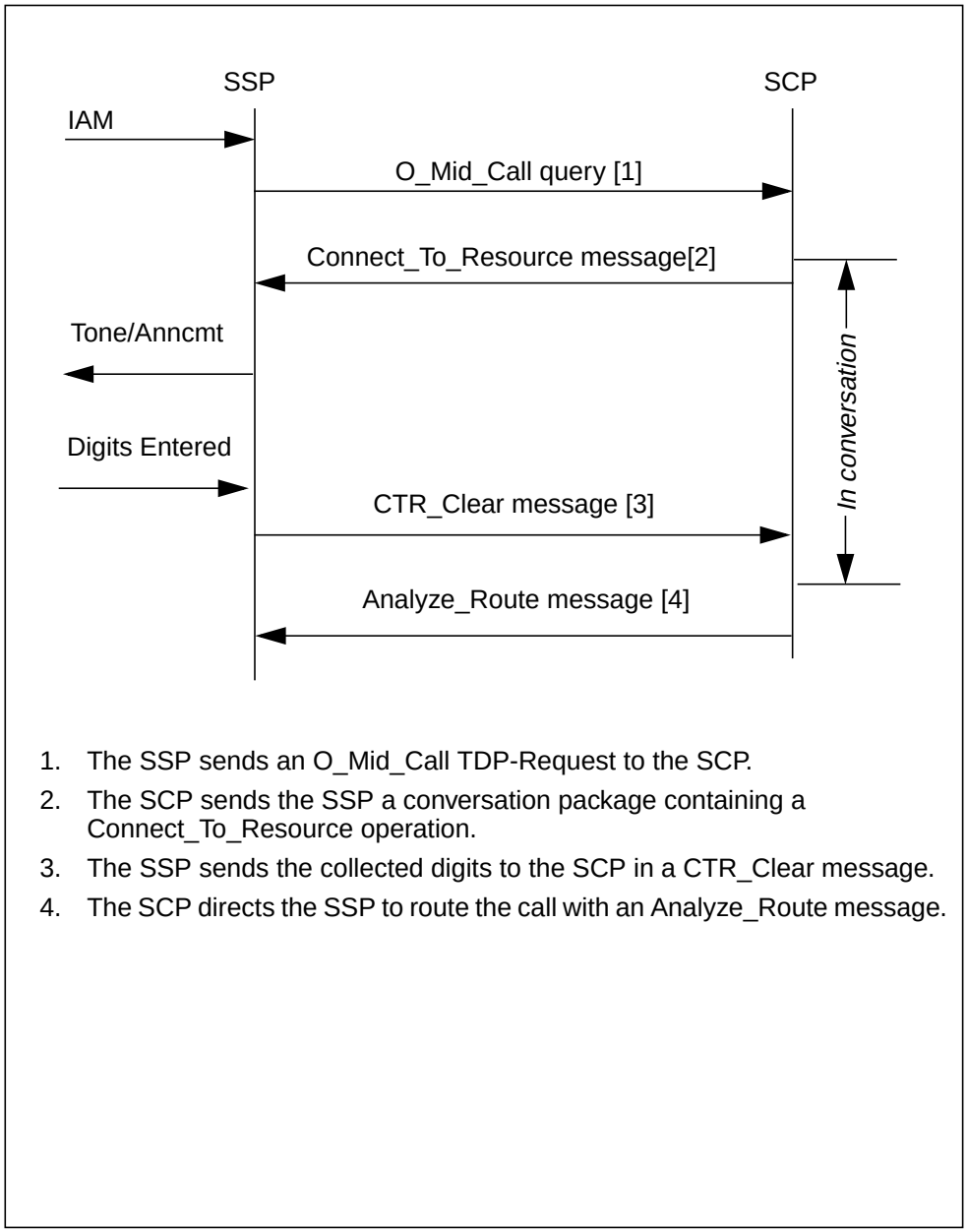


1. The SSP sends an Info_Collected TDP-Request to the SCP.
2. The SCP sends the SSP a conversation package containing a Send_To_Resource operation with a DestinationAddress parameter directing the SSP to start an IP connection.
3. The IP communicates information to the SCP through a Call_Info_From_Resource operation in a conversation package.
4. The SCP responds to the IP with a Call_Info_To_Resource message (in a Return Result Component) in a conversation package.
5. The IP connection is finished and the SSP sends a Resource_Clear message to the SCP indicating the end of the connection.
6. The SCP tells the SSP how to route the call with an Analyze_Route operation in a response package.

9.18 FGD call with Connect_To_Resource

In Figure 17, the SSP sends an O_Mid_Call TDP-Request to the SCP. The SCP sends the SSP a conversation package containing a Connect_To_Resource operation. The SSP returns the collected digits to the SCP in a CTR_Clear message. Finally the SSP directs the SSP to route the call with an Analyze_Route message.

Figure 17 FGD call with Connect_To_Resource

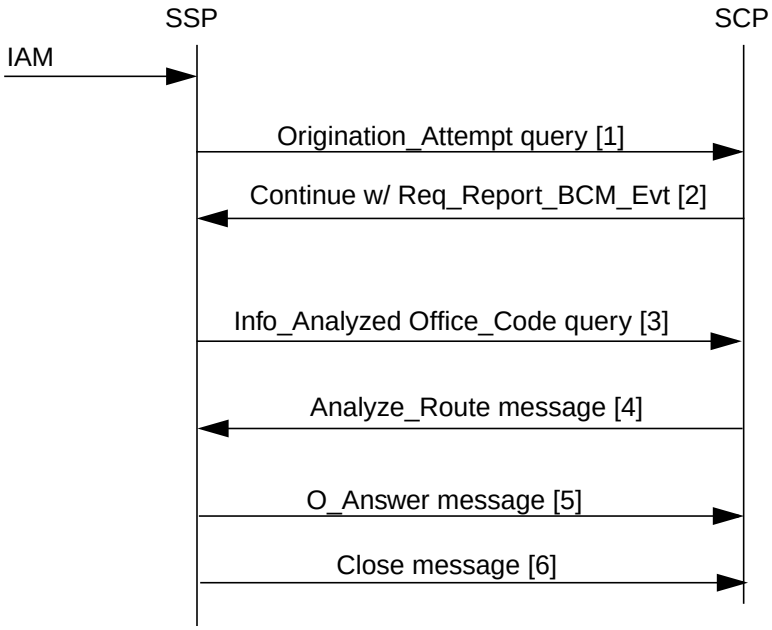


1. The SSP sends an O_Mid_Call TDP-Request to the SCP.
2. The SCP sends the SSP a conversation package containing a Connect_To_Resource operation.
3. The SSP sends the collected digits to the SCP in a CTR_Clear message.
4. The SCP directs the SSP to route the call with an Analyze_Route message.

9.19 SS7 FGD EDP call with Office_Code Query

In Figure 18, the SCP instructs the SSP to arm EDPs. When appropriate, the SSP sends an Office-Code query to the SCP. The SSP records the LNP information returned from the SCP and continues with EDP processing.

Figure 18 SS7 FGD EDP call with Office_Code Query

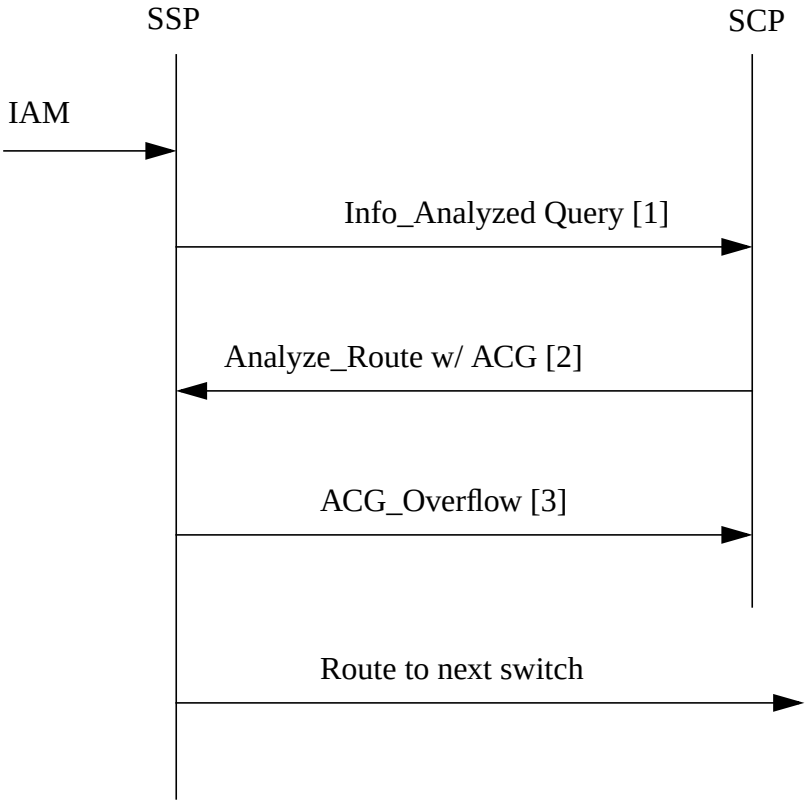


1. The SSP sends an Origination_Attempt TDP-Request to the SCP.
2. The SCP sends the SSP a conversation package containing a Continue component followed by a Request_Report_BCM_Event component which arms the O_Answer EDP.
3. The SSP sends an LNP Info_Analyzed TDP-Request to the SCP.
4. The SCP returns an Analyze_Route message to the SSP with updated LNP information.
5. After the terminator answers, the SSP sends an O_Answer EDP-Notification to the SCP in a conversation package. EDPs are not deactivated.
6. Since no more EDPs are reachable after O_Answer, the SSP sends a Close message to the SCP in a response package and deactivates EDPs.

9.20 ACG with ACG_Overflow

In Figure 19, the following the receipt of an ACG component, the SSP attempts to add the ACG control to the control list. Since the control list is full, the SSP generates an ACG_Overflow message in a unidirectional package to send to the SCP. The call is then routed according to the information received in the Analyze_Route component.

Figure 19 ACG with ACG_Overflow

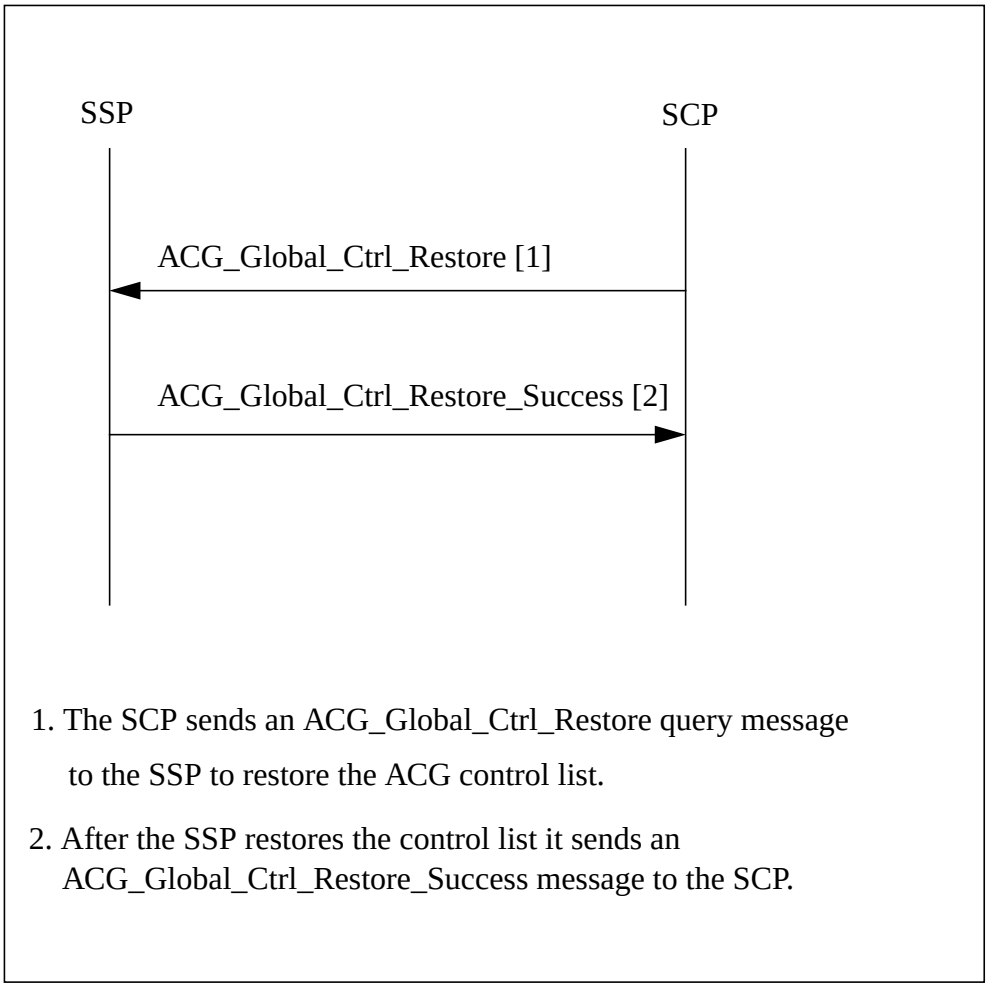


1. The SSP sends an Info_Analyzed query to the SCP.
2. The SSP receives a response package containing an Analyze_Route component followed by an ACG component which is to add an ACG control to the control list.
3. The SSP detects the control list to be in an overflow condition and sends an ACG_Overflow message to the SCP.

9.21 ACG_Global_Ctrl_Restore with ACG_Global_Ctrl_Restore_Success

In , the SCP sends an ACG_Global_Ctrl_Restore query message to restore the ACG control list. When the SSP receives the ACG_Global_Ctrl_Restore message it restores the control list and sends an ACG_Global_Ctrl_Restore_Success response message.

Figure 20 ACG_Global_Ctrl_Restore with ACG_Global_Ctrl_Restore_Success

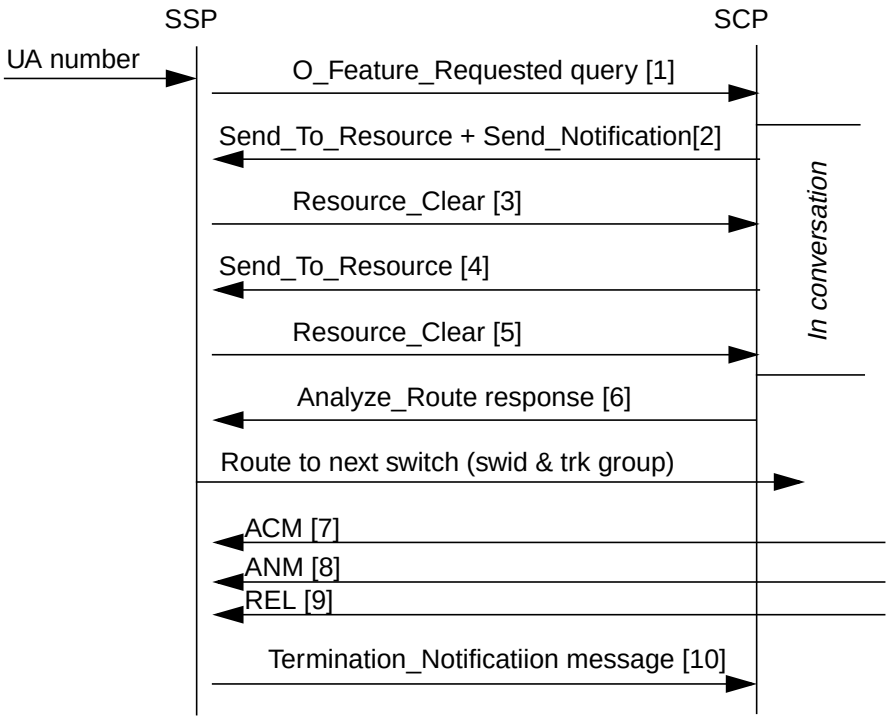


1. The SCP sends an ACG_Global_Ctrl_Restore query message to the SSP to restore the ACG control list.
2. After the SSP restores the control list it sends an ACG_Global_Ctrl_Restore_Success message to the SCP.

9.22 Termination_Notification with UA

In Figure 21, the customer is using a debit card. This customer may dial a Universal Access (UA) number which triggers at O_Feature_Requested. The SCP decides it needs additional information to complete the call (the users access code) and sends a Send_To_Resource (STR). The user complies by entering in their personal access code at which the SSP sends a Resource_Clear message containing the collected digits. The SCP decides it needs additional information once again and sends another STR with instructions to collect additional digits (the destination address). The user once again enters the needed information and the SSP sends a Resource_Clear message containing the collected digits. The call connects and the originating trunk waits for the call to be released. Once the call has been released the SSP sends the Termination_Notification message to the SCP.

Figure 21 Termination_Notification with UA



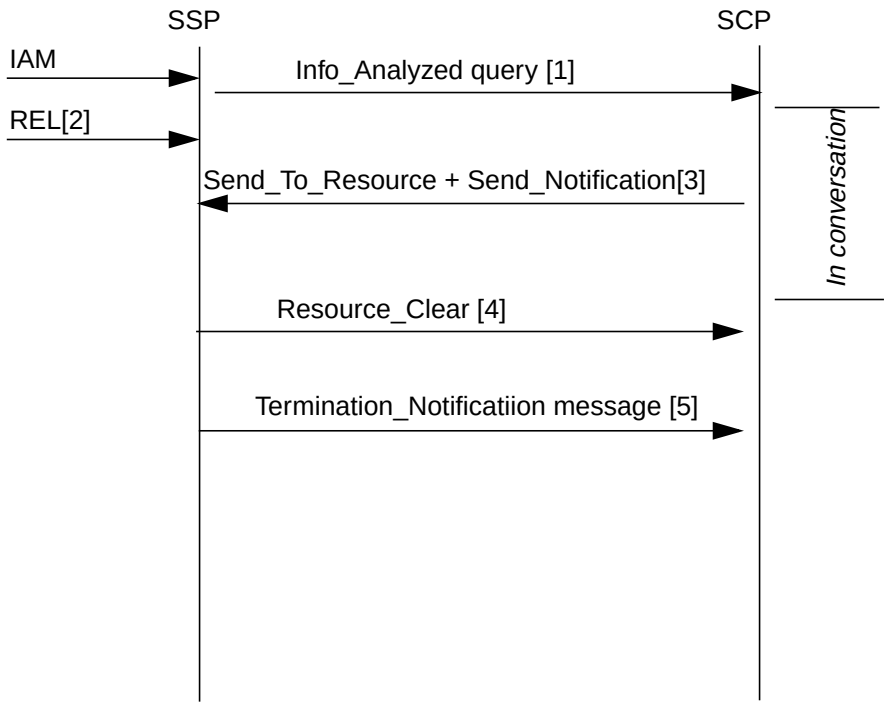
1. The SSP sends an O_Feature_Requested TDP-Request to the SCP.
2. The SCP sends the SSP a conversation package containing an STR component requesting to collect digits followed by a Send_Notification component.
3. The SSP sends a Resource_Clear message with the user entered digits is sent to the SCP for review.
4. The SCP sends the SSP another conversation package containing an STR component requesting to collect digits.
5. The SSP sends a Resource_Clear message with the user entered digits is sent to the SCP for review.
6. The SCP provides the following information in a (non-conversational) response package containing an Analyze_Route component only.
7. ACM is received signaling the called party has been notified. This is sent from the terminating trunk not the SCP.
8. ANM is received signaling the call has been answered. This is sent from the terminating trunk not the SCP.
9. REL is received signaling the call has been released.
10. The SSP sends a Termination_Notification message to the SCP.

The call duration 20 min 10.0 sec which is reflected in the CONNTIME parameter

9.23 Termination_Notification with REL message

In Figure 22,the originating SSP sends an Info_Analyzed query message to the SCP. While the SCP is decoding the query message and encoding a response message the originating SSP receives a REL message (probably because the calling party went on-hook). Meanwhile the SCP still believes the SSP is waiting for a response and sends a STR message in conversation with an extra Send_Notification component message. The SSP sends a Resource_Clear message which alerts the SCP that the call is finished and to close the STR transaction. The SSP sends a Termination_Notification message.

Figure 22 Termination_Notification with REL message



1. The SSP sends an Info_Analyzed TDP-Request to the SCP.
2. The SSP receives a REL message from the originator.
3. The SCP sends the SSP a conversation package containing an STR component requesting to collect digits followed by a Send_Notification component
4. The SSP sends a Resource_Clear message to SCP.
5. The SSP sends a Termination_Notification message to the SCP.

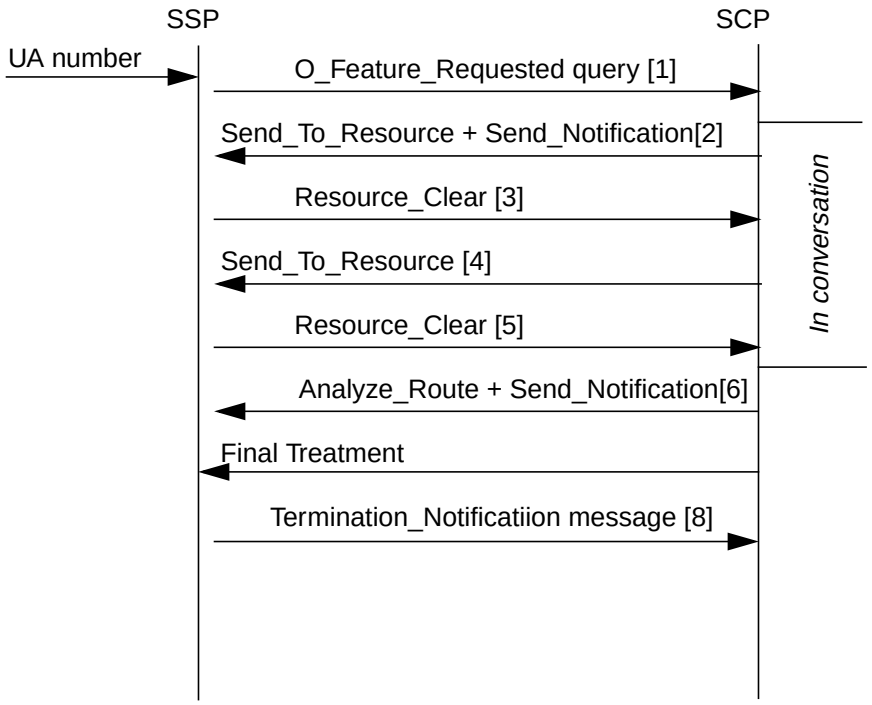
9.24 Termination_Notification with AINF

In Figure 23, the customer is using a debit card. This customer may dial a Universal Access (UA) number which triggers at O_Feature_Requested. The SCP decides it needs addition information to complete the call (the users access code) and sends a Send_To_Resource (STR). The user complies by entering in there personal access code at which the SSP sends a Resource_Clear message containing the collected digits. The SCP decide it needs additional information once again and sends another STR with instructions to collect additional digits (the destination address). The user once again enters the needed information and the SSP sends a Resource_Clear message containing the collected digits. The SSP receives the Analyze_Route message with another Send_Notification. It determines that the Analyze_Route message is incomplete (missing a mandatory parameter). The SSP invokes AIN Final Treatment but does not discard the rest of the message. The SSP send the Termination_Notification message to the SCP.

Note: *The last Send_Notification message is the one to which the SSP responds.*

Note: *Since the call did not connect there would not be a ConnectTime*

Figure 23 Termination_Notification with AINF



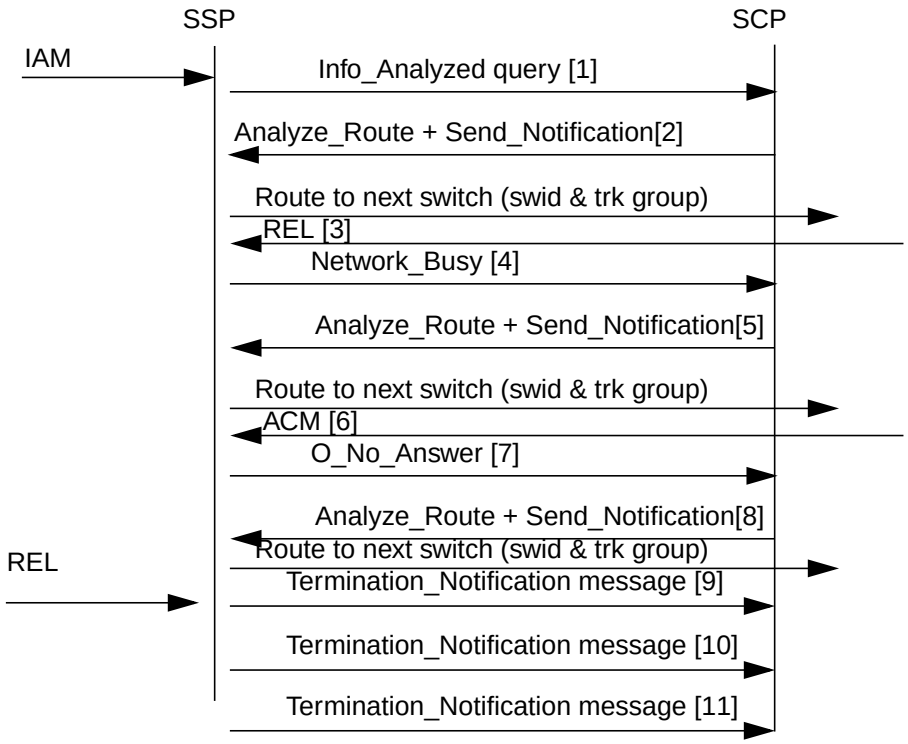
1. The SSP sends an O_Feature_Requested TDP-Request to the SCP.
2. The SCP sends the SSP a conversation package containing an STR component requesting to collect digits followed by a Send_Notification component
3. The SSP sends a Resource_Clear message with the user entered digits is sent to the SCP for review
4. The SCP sends the SSP another conversation package containing an STR component requesting to collect digits
5. The SSP sends a Resource_Clear message with the user entered digits is sent to the SCP for review
6. The SCP provides the following information in a response package containing an Analyze_Route component and a Send_NotifIcation component.
7. The SSP detects a need to perform the AIN Final treatment action
8. The SSP sends a Termination_Notification message to the SCP.

9.25 Multiple Send_Notification with Termination_Notification

In Figure 24, The originating switch receives an IAM message. The called number triggers at Info_Analyzed. The SCP sends an Analyze_Route message with detail instructions on routing the call plus a second component containing a Send_Notification message. The destination switch is busy and sends back a REL message with a cause value of no circuit available. The originating SSP triggers at Network_Busy and receives another Analyze_route with another Send_Notification. This time the destination switch is not busy and sends an ACM message back to the originating switch. The called party has not answered and the O_No_Answer timer has expired. The originating switch triggers again at O_No_Answer. It does not release the resources until it receives another route. After receiving the final Analyze_Route with another Send_Notification component the calling party goes on-hook.The SSP sends three Termination_Notification messages. Each one containing a different ECHODATA and TERMIND value.

Note: *Although each Send_Notification message contains the same information except for the ECHODATA parm, each Send_Notification message could go to a different SCP.*

Figure 24 Multiple Send_Notification with Termination_Notification

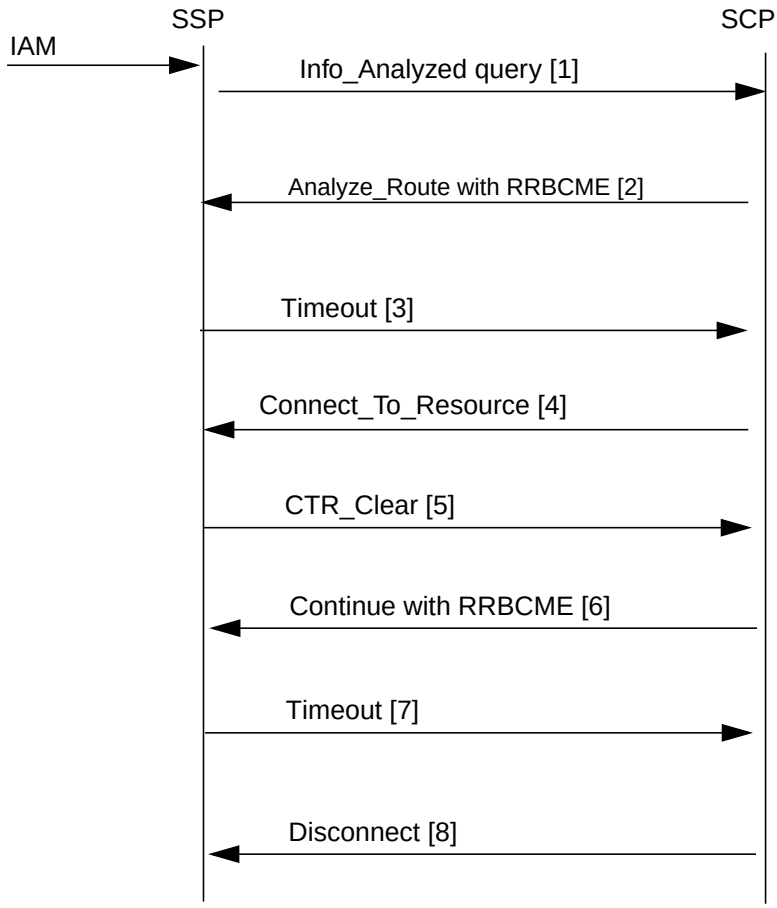


1. The SSP sends an Info_Analyzed TDP-Request to the SCP.
2. The SCP sends the SSP an Analyze_Route component followed by a Send_Notification component
3. The SSP receives a REL message (network_busy).
4. The SSP sends a Network_Busy TDP-Request to the SCP.
5. The SCP sends the SSP an Analyze_Route component followed by a Send_Notification component.
6. The SSP receives an ACM message signaling the called party has been notified.
7. The SSP O_No_Answer timer has expired. It sends a O_No_Answer TDP-Request to the SCP.
8. The SCP sends the SSP an Analyze_Route component followed by a Send_Notification component.
9. REL is received (by the called party) signaling the call has been released.
10. The SSP sends a Termination_Notification message to the SCP.
11. The SSP sends a Termination_Notification message to the SCP.
12. The SSP sends a Termination_Notification message to the SCP

9.26 Connect_To_Resource from Timeout EDP-Request

In Figure 25, the SCP replies to an Info_Analyzed query by sending a conversation package containing an Analyze_Route call-related operation and a RRBCME (Request Report BCM Event) operation arming the Timeout event. After the terminator answers, the TimeoutTimer is started. When the timer expires, the switch sends a Timeout EDP-Request to the SCP, and the SCP responds with a Connect_To_Resource message in a conversation package. The SSP connects the specified party or parties to the resource and sends a conversational CTR_Clear message to the SCP once the resource is completed. The SCP responds by sending the SSP a conversation package containing a Continue operation and a RRBCME operation which requests the Timeout event again. At this point, the SCP starts the TimeoutTimer again and allows the calling and called parties to remain connected. When the TimeoutTimer expires the second time, the SSP sends another Timeout EDP-Request to the SCP. This time, the SCP sends the SSP a Disconnect message in a response package. The SSP idles the called party and sends the calling party to treatment.

Figure 25 Connect_To_Resource from Timeout EDP-Request



1. The SSP sends an Info_Analyzed TDP-Request to the SCP.
2. The SSP receives an Analyze_Route with a RRBCME operation arming the Timeout event.
3. The TimeoutTimer expires and the SSP sends a Timeout EDP-Request to the SCP.
4. The SCP responds with a Connect_To_Resource message in a conversation package.
5. The SSP connects the specified party or parties to the resource and sends a conversational CTR_Clear message to the SCP once the resource is completed.
6. The SCP responds by sending the SSP a conversation package containing a Continue operation and a RRBCME operation which requests the Timeout event again.(The SCP starts the TimeoutTimer again and allows the calling and called parties to remain connected.)
7. The TimeoutTimer expires the second time, the SSP sends another Timeout EDP-Request to the SCP.
8. The SCP sends the SSP a Disconnect message in a response package. The SSP idles the called party and sends the calling party to treatment.

9.27 Call Transfer via SwitchHookFlash (Scenario 1)

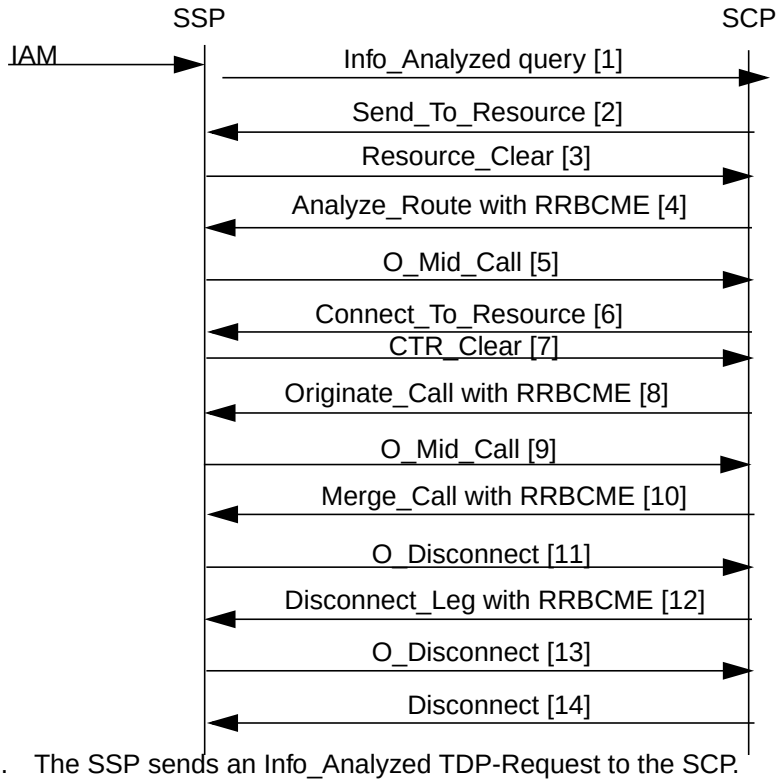
In Figure 26, the originator places a call which triggers at the Info_Analyzed PIC. After performing SCP driven digit collection with a Send_To_Resource message, the SCP directs the SSP to route the call to an operator center with an analyze_route message which also arms the SwitchHookFlash EDP. Upon answer, the operator realizes that the originator has reached the wrong extension. In order to transfer the originator to the correct extension, the operator enters an asterisk (‘*’) which causes an O_Mid_Call message to be sent to the SCP. The SCP responds with a Connect_To_Resource message. Upon receipt of the Connect_To_Resource message, all users indicated in the message are connected to the resource. In this case only the operator is connected to the resource while the originator hears silence. The operator interacts with the resource and indicates whom s/he wants to conference into the call. The digits collected by the SSP are returned to the SCP in the CTR_Clear message. The SCP parses the CTR_Clear message and sends back an Originate_Call message which provides the SSP with the routing parameters required to reach the third party. The Originate_Call message also contains a Request_Report_BCM_Event component which re-arms the SwitchHookFlash EDP.

The second call segment routes with the information contained in the Originate_Call message. When the third party answers, the operator enters an asterisk (‘*’), again causing the SSP to send an O_Mid_Call message to the SCP. The SCP directs the SSP to join all three parties with a Merge_Call message which also arms the oDisconnect event as a request.

The operator disconnects. The SSP then sends an O_Disconnect message to the SCP indicating that the operator has disconnected. The SCP responds with a Disconnect_Leg message which removes the specified party from the call and also re-arms the oDisconnect event as a request.

The TCAP transaction stays open until one of the remaining parties disconnects. Regardless of which party disconnects, the SSP sends an O_Disconnect message. The SCP responds with a Disconnect message, taking down the call.

Figure 26 Call Transfer via SwitchHookFlash (Scenario 1)



1. The SSP sends an Info_Analyzed TDP-Request to the SCP.
2. The SSP receives a Send_To_Resource instructing it to collect digits.
3. The SSP sends a Resource_Clear containing the collected digits.
4. The SCP responds with an Analyze_Route arming the SHF event.
5. The SSP detects the asterisk key being pressed for .5 seconds and sends an O_Mid_Call message to the SCP.
6. The SCP responds by sending a Connnect_To_Resource message to the SSP instructing it to collect the address digits of the third party.
7. The SSP sends a Resource_Clear message containing the address digits.
8. The SCP responds with an Originate_Call message instructing the SSP to create another call segment and arm the O_Disconnect, O_Abandon, and SHF events.
9. The SSP then sends an O_Mid_Call (SHF) message as a result of the operator pressing the ‘*’ key.
10. The SCP responds with a Merge_Call message instructing the SSP to merge all three legs of the call and rearming the O_Disconnect, O_Abandon, and SHF events.
11. The SSP detects that the operator went on hook and sends the O_Disconnect message to the SCP.
12. The SCP responds with a Disconnect_Leg messaging instructing the SSP to disconnect the leg and rearm the O_Disconnect event.
13. The SSP sends another O_Disconnect message when one of the remaining parties goes on hook.
14. The SCP resonds with a Disconnect message, bringing down the call.

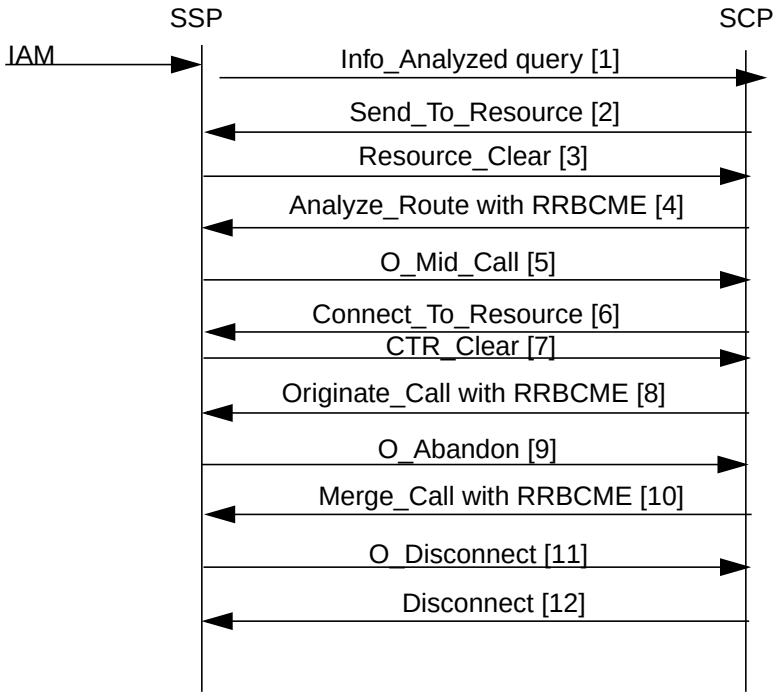
9.28 Call Transfer via SwitchHookFlash (Scenario 2)

In , the originator places a call which triggers at the Info_Analyzed PIC. After performing SCP driven digit collection with a Send_To_Resource message, the SCP directs the SSP to route the call to an operator center with an analyze_route message which also arms the SwitchHookFlash EDP. Upon answer, the operator realizes that the originator has reached the wrong extension. In order to transfer the originator to the correct extension, the operator enters an asterisk (‘*’) which causes an O_Mid_Call message to be sent to the SCP. The SCP responds with a Connect_To_Resource message. Upon receipt of the Connect_To_Resource message, all users indicated in the message are connected to the resource. In this case only the operator is connected to the resource while the originator hears silence. The operator interacts with the resource and indicates whom s/he wants to conference into the call. The digits collected by the SSP are returned to the SCP in the CTR_Clear message. The SCP parses the CTR_Clear message and sends back an Originate_Call message which provides the SSP with the routing parameters required to reach the third party. The Originate_Call message also contains a Request_Report_BCM_Event component which re-arms the SwitchHookFlash EDP.

The second call segment routes with the information contained in the Originate_Call message. Before the third party answers the call, however, the operator goes on hook. This causes an O_Abandon message to be sent to the SCP. The SCP responds with a Merge_Call message which causes the originator and third party to be merged together on one call. The Merge_Call message also arms the O_Disconnect event.

The TCAP transaction stays open until one of the remaining parties disconnects. Regardless of which party disconnects, the SSP sends an O_Disconnect message. The SCP responds with a Disconnect message, taking down the call.

Call Transfer via SwitchHookFlash (Scenario 2)



1. The SSP sends an Info_Analyzed TDP-Request to the SCP.
2. The SSP receives a Send_To_Resource instructing it to collect digits.
3. The SSP sends a Resource_Clear containing the collected digits.
4. The SCP responds with an Analyze_Route arming the SHF event.
5. The SSP detects the asterisk key being pressed for .5 seconds and sends an O_Mid_Call message to the SCP.
6. The SCP responds by sending a Connnect_To_Resource message to the SSP instructing it to collect the address digits of the third party.
7. The SSP sends a Resource_Clear message containing the address digits.
8. The SCP responds with an Originate_Call message instructing the SSP to create another call segment and arm the O_Disconnect, O_Abandon, and SHF events.
9. The operator goes on-hook, causing an O_Abandon message to be sent to the SCP.
10. The SCP responds with a Merge_Call message instructing the SSP to merge the two remaining legs of the call and rearm the O_Disconnect event.
11. The SSP sends an O_Disconnect message when one of the remaining parties goes on hook.
12. The SCP resonds with a Disconnect message, bringing down the call.

Appendix A: Parameter encodings

A.1 Introduction

This appendix provides the encoding rules for all extension parameters defined in this document, as well as any extended indicators used in parameters (for example, additional Nature of Address indicators). All messages and parameters which are used as described in Bellcore specification GR-1299-CORE are not listed here; see GR-1299-CORE for their encodings.

The encodings below are presented using an abstract syntax notation known as ASN.1. For additional information on ASN.1, see CCITT Recommendations X.208 and X.209. For additional information on TCAP, see Bellcore specification GR-246-CORE.

A.2 Encodings for AIN parameters

A.2.1 Encoding the Numbering Plan values

RA-1 **[146]** The encodings shown in Table 9-94 shall be used for the Numbering Plan indicator in digits parameters.

Table 9-94 Encodings for Numbering Plan

Name	Value in hex
unknown	0
isdn (Rec E.164)	01
private	05

A.2.2 Encoding the Nortel specific Nature of Address values for Called and Calling Parameters

RA-2 **[147v4]** The encodings shown in Table 9-95 shall be used for the Nature of Address indicator for Called and Calling Parameters. The

values shown are in addition to those of GR-1299-CORE [207] and [208].

Table 9-95 Encodings for Nature of Address in Called and Calling Parameters

Name	Value in hex
partitioned	60
acct	61
ani	62
i2ani	63
authcode	64
hotline	65
mccs	66
pin	67
vpn	68
n00	69

A.2.3 Encoding the Nortel specific Nature of Address values for Charge Number Parameters

RA-3 **[865]** The encodings shown in Table 9-96 shall be used for the Nature of Address indicator for Charge Number parameters. The values shown are in addition to those of GR-1299-CORE [209].

Table 9-96 Encodings for Nature of Address in Charge Number Parameters

Name	Value in hex
unknown	00
international	04
ani	62
i2ani	63
authcode	64
mccs	66
n00	69

A.2.4 Encoding the assignment authority

See section 2.5.4 for more information on the assignment authority field in extension parameter and FlexParameterBlock sets.

RA-4 **[148v2]** The following hex value shall be used as the assignment authority in the `ExtensionParameter` parameter and the `FlexParameterBlock` to identify the parameter set described in Chapter 8, “Extension parameters”. This value shall be encoded in nine octets:

```
2A 86 48 86 F6 7D 08 41 10
```

RA-5 **[871]** The following hex value shall be used as the assignment authority in the `ExtensionParameter` parameter to identify the parameter set described in Chapter 8, “Extension parameters” for all extension parameters which belong to the `infoCollectedExtensionAa` family. This value shall be encoded in thirteen octets:

```
60 86 48 01 86 F7 09 04 01 01 81 C8 02
```

RA-6 **[149]** The `parameters` field of the `ExtensionParameter` parameter using the above assignment authorities shall have the format of an ASN.1 `SET` of the parameters described in Chapter 8, “Extension parameters”.

A.2.5 Encoding the trigger criteria type

RA-7 **[150v4]** The spare values shown in Table 9-97 shall be used to encode the criteria type for various triggers.

Table 9-97 Encodings for the trigger criteria type (page 1 of 2)

Name	Value in hex
csp_clid	60
csp_addr	61
csp_n00	62
csp_intl	63
sds_info	64
sds_ani	65
sds_n00	66
sds_cic	67
sio_cic	68
sio_info	69

Name	Value in hex
sio_ani	6A
sio_addr	6B
sio_n00	6C
sio_intl	6D
csp_adin	6E
sds_adin	6F
sio_adin	70
oIECReorigination	71

A.2.6 Encoding generic address Type of Digits values

RA-8 **[151v2]** The values shown in Table 9-98 shall be used to encode the Type of Digits for the `GenericAddress` parameter.

Table 9-98 Encodings for Type of Digits

Name	Value in hex
AlternateOutpulseNo	60
SecondAlternateOutpulseNo	61
OverflowRoutingNo	62
LNPPortedDialedNo	C0
DNIS	00

A.2.7 Encoding maximum digits to collect values

RA-9 **[302]** Spare values shall be used within the `MaximumDigits` field of the `AnnouncementDigitBlock` to indicate the maximum number of digits to collect as part of a variable-length digit string. The number of digits shall be added to a base value of 128 to be encoded. As a result, the field is defined as follows:

```
MaximumDigits ::= INTEGER (0..255)
-- range:   0-- 24 is “fixed number of digits”
-- range:  25--127 is spare
-- range: 128--152 is “up to max number of digits”
-- range: 153--252 is spare
-- 253 is “normal number of digits”
-- 254 is “variable number of digits”
-- 255 is spare
```

Thus, for example, an indication to collect up to 12 digits would be encoded using the value 140 (128 + 12). Note also that the "fixed" range of digits is limited to 24, as that is the most the DMS-250 will collect and report.

A.2.8 Encoding the EDPNotification parameter

The `EDPNotification` parameter is described in section 6.62 of GR-1299-CORE.

RA-10 **[756]** The `EDPNotification` parameter shall be encoded as follows:

```
EDPNotification ::= [93] IMPLICIT BIT STRING {
   oCalledPartyBusy (0), -- always "0"
   oNoAnswer(1), -- always "0"
   oTermSeized(2),
   oAnswer(3),
   tBusy(4), -- always "0"
   tNoAnswer(5), -- always "0"
   termResourceAvailable(6),
   tAnswer(7),
   networkBusy(8), -- always "0"
   oSuspended(9), -- always "0"
   oDisconnectCalled(10), -- always "0"
   oDisconnect(11),
   oAbandon(12), -- always "0"
   featureActivator(13), -- always "0"
   switchHookFlash(14), -- always "0"
   success(15), -- always "0"
   tDisconnect(16), -- always "0"
   timeout(17), -- always "0"
   originationAttempt(18)
   }
```

A.2.9 Encoding the EDPRequest Parameter

The `EDPRequest` parameter is described in section 6.63 of GR-1299-CORE.

RA-11 **[757]** (Deleted)

A.3 Encoding the O_Feature_Requested message

The `O_Feature_Requested` message is not defined in existing Bellcore specifications (for example, GR-1299-CORE), but is added here in a manner consistent with the treatment of the `Info_Collected` message.

Operation	Family	Specifier	Class[a]	Hex Value
Request/Notification — O_Feature_Requested	01100100	01100000	Class 2	6460

a. Class: (2) only failure is reported

RA-12 **[303v2]** The following ASN.1 definition shall be used for the `O_Feature_Requested` message:

```
oFeatureRequested OPERATION
PARAMETER         SEQUENCE {
                  UserID,
                  BearerCapability,
                  ChargeNumber,            OPTIONAL
                  Lata,                    OPTIONAL
                  Carrier,                 OPTIONAL
                  TriggerCriteriaType,
                  PointInCall,
                  FeatureActivatorID,
                  CallingPartyID,          OPTIONAL
                  ChargePartyStationType,  OPTIONAL
                  AccessCode,              OPTIONAL
                  CollectedAddressInfo,    OPTIONAL
                  CollectedDigits,         OPTIONAL
                  VerticalServiceCode,     OPTIONAL
                  GenericAddressList,      OPTIONAL
                  ExtensionParameter       OPTIONAL
                  }
ERRORS {          applicationError,
                  failureReport
                  }
-- LINKED {       -- analyzeRoute,
                  -- disconnect,
                  -- sendToResource
                  -- }

::= private: 25696
-- Family Name is “Request/Notification”
```

A.4 Encodings for extension parameters in the NT_DMS250_EXTENSION_PARAMETER_FAMILY

A.4.1 Encoding the STS pair

RA-13 **[152]** The `servTranslationScheme` extension parameter (section 8.4.1) shall use the following format:

```
servTranslationScheme ::=
            [1] IMPLICIT INTEGER (0..999)
```

A.4.2 Encoding the primaryTrunkGroupSTS parameter

RA-14 **[619]** The `primaryTrunkGroupSTS` parameter (section 8.4.1) shall use the following format:

```
primaryTrunkGroupSTS ::= [32] IMPLICIT INTEGER (0..999)
```

A.4.3 Encoding the alternateTrunkGroupSTS parameter

RA-15 **[620]** The `alternateTrunkGroupSTS` parameter (section 8.4.1) shall use the following format:

```
alternateTrunkGroupSTS ::= [33] IMPLICIT INTEGER (0..999)
```

A.4.4 Encoding the secondAlternateTrunkGroupSTS parameter

RA-16 **[621]** The `secondAlternateTrunkGroupSTS` parameter (section 8.4.1) shall use the following format:

```
secondAlternateTrunkGroupSTS ::= [34] IMPLICIT INTEGER
                                      (0..999)
```

A.4.5 Encoding the overflowRoutingNoSTS parameter

RA-17 **[622]** The `overflowRoutingNoSTS` parameter (section 8.4.1) shall use the following format:

```
overflowRoutingNoSTS ::= [35] IMPLICIT INTEGER (0..999)
```

A.4.6 Encoding the call type

RA-18 **[153v2]** The `callType` parameter (section 8.4.2) shall use the following format:

```
callType ::= [2] IMPLICIT ENUMERATED {
   offnet                 (0),
   onnet                  (1),
   forced_onnet           (2),
   virtual_onnet          (3)
}
```

A.4.7 Encoding the satellite restriction

RA-19 **[154]** The `satRestriction` parameter (section 8.4.3) shall use the following format:

```
satRestriction ::= [3] IMPLICIT NULL
-- presence = do not route via satellite
-- absence  = no restriction
```

A.4.8 Encoding the class of service

RA-20 **[155v2]** The `classOfSvc` parameter (section 8.4.4) shall use the following format:

```
classOfSvc ::= [4] IMPLICIT INTEGER (0..2047)
```

A.4.9 Encoding the billing sequence number

RA-21 **[156]** The `billSequenceNumber` parameter (section 8.6.1) shall use the following format:

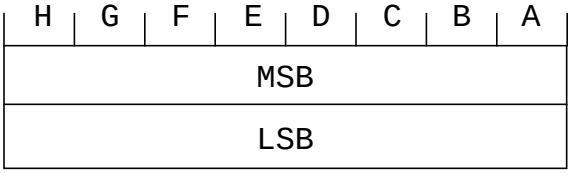
```
billSequenceNumber ::=
            [9] IMPLICIT OCTET STRING (SIZE(4))
```

A.4.10 Encoding the call branding announcement

RA-22 **[157]** The `callBranding` parameter (section 8.7.1) shall use the following format:

```
callBranding ::= [5] IMPLICIT OCTET STRING (SIZE(2))
```

The value of the announcement identifier shall be filled as shown here:

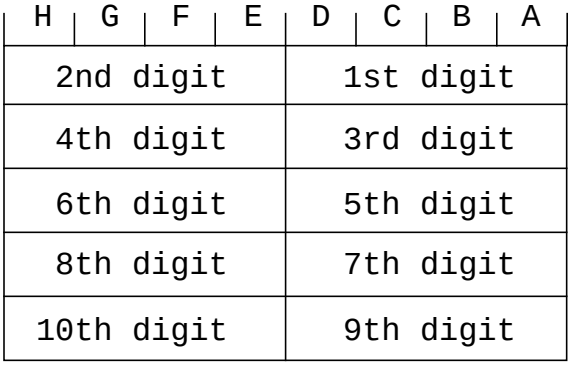


A.4.11 Encoding the Universal Access number

RA-23 **[158v2]** The `universalAccess` parameter (section 8.8.1) shall use the following format *when the version designation of the CAIN protocol version is* `V0`:

```
universalAccess ::= [10] IMPLICIT OCTET STRING (SIZE(5))
```

The digits of the Universal Access number shall be filled as shown here:



RA-24 **[304]** *When the version designation of the CAIN protocol version* is `V1` or later, the following format shall be used (based on the `AINDigits` parameter format from GR-1299-CORE):

```
universalAccess ::= [10] IMPLICIT AINDigits
```

A.4.12 Encoding the connect to SCU indicator

RA-25 **[305]** The `connectToSCU` parameter (section 8.9.1) shall use the following format:

```
connectToSCU ::= [11] IMPLICIT NULL
-- presence = connect this call to the PSN SCU
-- absence  = no action
```

A.4.13 Encoding the CAIN subscription group

RA-26 **[306]** The `cainGroup` parameter (section 8.11.1) shall use the following format:

```
cainGroup ::= [12] IMPLICIT INTEGER (0..4095)
```

A.4.14 Encoding the authcode database index

RA-27 **[307]** The `adin` parameter (section 8.5.1) shall use the following format:

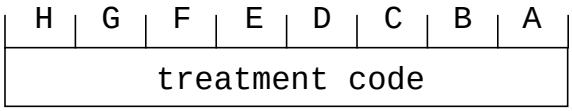
```
adin ::= [13] IMPLICIT INTEGER (0..99)
```

A.4.15 Encoding the treatment code

RA-28 **[308]** The `treatment` parameter (section 8.10.1) shall use the following format:

```
treatment ::= [15] IMPLICIT OCTET STRING (SIZE (1))
```

The value of the treatment code shall be filled as shown here:



A.4.16 Encoding the reoriginated call indicator

RA-29 **[309]** The `reorigCall` parameter (section 8.12.5) shall use the following format:

```
reorigCall ::= [16] IMPLICIT NULL
-- presence = call is a reoriginated call
-- absence  = no indication
```

A.4.17 Encoding the reorigination allowed indicator

RA-30 **[310]** The `reorigAllowed` parameter (section 8.13.1) shall use the following format:

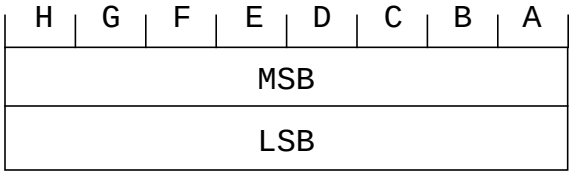
```
reorigAllowed ::= [18] IMPLICIT INTEGER (0..99)
```

A.4.18 Encoding the trunk information

RA-31 **[311]** The originating and terminating trunk information parameters use the following data definition:

```
trunkInfo ::= SEQUENCE {
   trunkType      [0] IMPLICIT ENUMERATED {
                        dal         (0),
                        dal-tie     (1),
                        eant        (2),
                        pri         (3),
                        imt         (4),
                        axxess      (5),
                        fg-a        (6),
                        fg-b        (7),
                        fg-c        (8)
                   },
   trunkGroup     [1] IMPLICIT INTEGER (0..9999),
   trunkMember    [2] IMPLICIT OCTET STRING (SIZE(2))
}
```

The value of the trunk member shall be filled as shown here:



RA-32 **[312]** The `origTrunkInfo` parameter (section 8.12.1) shall use the following format:

```
origTrunkInfo ::= [14] IMPLICIT trunkInfo
```

RA-33 **[313]** The `termTrunkInfo` parameter (section 8.12.3) shall use the following format:

```
termTrunkInfo ::= [17] IMPLICIT trunkInfo
```

A.4.19 Encoding the busy route information

RA-34 **[314]** The `busyRoute` parameter (section 8.12.4) shall use the following format:

```
busyRoute ::= [8] IMPLICIT SEQUENCE {
   numberOfTries  [0] IMPLICIT INTEGER (1..99),
   routingNo      [1] IMPLICIT AINDigits, OPTIONAL
   routeIndex     [2] IMPLICIT OCTET STRING (SIZE(4))
                         OPTIONAL
}
```

The route index shall be 8 digits, encoded in BCD as shown here:

H	G	F	E	D	C	B	A
2nd digit				1st digit			
4th digit				3rd digit			
6th digit				5th digit			
8th digit				7th digit			

A.4.20 Encoding the Network_Busy EDP actions

RA-35 **[423]** The `networkBusyActions` parameter (section 8.14.1) shall use the following format:

```
networkBusyActions ::=
            [19] IMPLICIT OCTET STRING (SIZE(1))
```

H	G	F	E	D	C	B	A
Spare			TERMRTE_GNCT		RTESDONE	RTEAVAIL	

RA-36 **[424]** The Route Available (RTEAVAIL) field shall be encoded as follows:

00 Request

01 Ignore

10 Next Route

11 Reserved

RA-37 **[425]** The Routes Done (RTESDONE) field shall be encoded as follows:

0 Request

1 Ignore

RA-38 **[426]** The Terminating Route Generalized No Circuit (TERMRTE_GNCT) field shall be encoded as follows:

00 Request

01 Ignore

10 Next route

11 Next CAIN route

A.4.21 Encoding the O_Called_Party_Busy EDP actions

RA-39 **[427]** The `oCalledPartyBusyActions` parameter (section 8.14.2) shall use the following format:

```
oCalledPartyBusyActions ::=
            [20] IMPLICIT OCTET STRING (SIZE(1))
```

H	G	F	E	D	C	B	A
Spare					RTESDONE	RTEAVAIL	

RA-40 **[428]** The Route Available (RTEAVAIL) field shall be encoded as follows:

00 Request

01 Ignore

10 Next Route

11 Next CAIN Route

RA-41 **[429]** The Routes Done (RTESDONE) field shall be encoded as follows:

0 Request

1 Ignore

A.4.22 Encoding the O_No_Answer EDP actions

RA-42 **[430]** The `oNoAnswerActions` parameter (section 8.14.3) shall use the following format:

```
oNoAnswerActions ::=
            [21] IMPLICIT OCTET STRING (SIZE(1))
```

H	G	F	E	D	C	B	A
Spare					RTESDONE	RTEAVAIL	

RA-43 **[431]** The Route Available (RTEAVAIL) field shall be encoded as follows:

00 Request

01 Ignore

10 Next Route

11 Next CAIN Route

RA-44 **[432]** The Routes Done (RTESDONE) field shall be encoded as follows:

0 Request

1 Ignore

A.4.23 Encoding the EDP digit buffer indicator

RA-45 **[433]** The `edpBuffer` parameter (section 8.14.4) shall use the following format:

```
edpBuffer ::= [22] IMPLICIT NULL
-- presence = buffer digits when sending EDP-R
-- absence  = do not buffer digits
```

A.4.24 Encoding the Universal Translations index

RA-46 **[306]** The `univIdx` parameter (section 8.16.1) shall use the following format:

```
univIdx ::= [23] IMPLICIT INTEGER (0..15)
```

A.4.25 Encoding the Multi-switch Business Group (MBG) NETINFO parameter

RA-47 **[434]** The `netinfo` parameter (section 8.15.1) shall use the following format:

```
netinfo ::= [24] IMPLICIT OCTET STRING (SIZE(7))
```

H	G	F	E	D	C	B	A
Spare	AttSt	BGID	LP II	Party Selector			
msb[a]	EXTNETID (2 octets)						
							lsb[b]
msb	NETCGID (2 octets)						
							lsb
msb	NCOS (2 octets)						
							lsb

a. msb = most significant bit

b. lsb = least significant bit

RA-48 **[435]** The Attendant Status (AttSt), Business Group Identifier (BGID) type, Line Privileges Information Indicator (LP II), and Party Selector fields shall use the encodings specified for the `BusinessGroup` parameter in section 6.30 of GR-1299-CORE.

The External Network ID (EXTNETID) is a 2-octet field which maps to a network name in table `NETNAMES` on the SSP.

RA-49 **[436]** The EXTNETID field shall be encoded as follows:

0000000000000000

to Assigned External Network codes

1111111111111111

The Network Customer Group ID (NETCGID) is a 2-octet field which maps to a unique network customer group name in table CUSTNTWK on an SL-100.

RA-50 **[437]** The NETCGID field shall be encoded as follows:

0000000000000000	
to	Assigned Network Customer Group codes
0000111111111111	
0001000000000000	
to	Spare
1111111111111111	

The Network Class of Service (NCOS) provides SL-100 customer group agents with various call completion priorities. NCOS values are uniquely assigned by customer groups and are used to determine call privileges for calls using the network. The NCOS is encoded as a 2-octet field.

RA-51 **[438]** The NCOS field shall be encoded as follows:

0000000000000000	
to	Assigned Network Class of Service codes
0000000111111111	
0000001000000000	
to	Spare
1111111111111111	

A.4.26 Encoding the T1 timeout overflow re-query indicator

RA-52 **[439]** The t1Overflow parameter (section 8.12.7) shall use the following format:

```
t1Overflow ::= [25] IMPLICIT NULL
-- presence = the current query message is a T1 timeout
              overflow re-query message
-- absence  = no indication
```

A.4.27 Encoding the Local Number Portability (LNP) information received indicator

RA-53 **[440]** The lnpReceived parameter (section 8.12.8) shall use the following format:

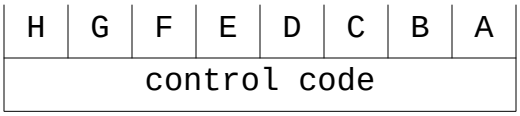
```
lnpReceived ::= [26] IMPLICIT NULL
-- presence = LNP information received
-- absence  = no indication
```


Nortel Networks Proprietary and Confidential

A.4.28 Encoding the Call Control parameter

RA-54 **[441]** The `callCtrl` parameter (section 8.12.8) shall use the following format:

```
callCtrl ::= [27] IMPLICIT OCTET STRING (SIZE(1))
```



RA-55 **[442]** The control code field shall be encoded as follows:

00000000	Unrestricted
00000001	Leave the current TDP (LEAVE_TDP)
00000010	Continue, do not trigger (CONT_NOTRIG)
00000011	
to	Reserved
11111111	

A.4.29 Encoding the amaDigits parameter

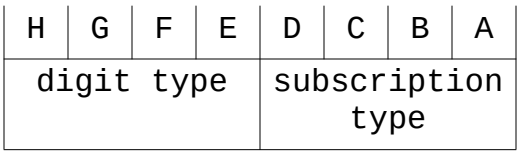
RA-56 **[623]** The `amaDigits` parameter (section 8.6.2) shall use the following format:

```
amaDigits ::= [37] Sequence (SIZE 1..6)) OF
                   AMADigitsDialedWC
```

A.4.30 Encoding the subscriptionInfo parameter

RA-57 **[624]** The `subscriptionInfo` parameter (section 8.11.3) shall use the following format:

```
subscriptionInfo ::= [28] IMPLICIT OCTET STRING
                          (SIZE (1))
```



RA-58 **[625]** The digit type field shall be encoded as follows:

```
0000     info
0001     adin
0010     ani
0011     xlaaddr
0100     addr
```

```
0101     cic
0110
 to      reserved
1111
```

RA-59 **[626v2]** The subscription type field shall be encoded as follows:

```
0000     SCP
0001     addr
0010     auth
0011     ani
0100     agent
0101     office
0110     tr533cic
0111     carrier
1000
 to      reserved
1111
```

A.4.31 Encoding the collectedAddress parameter

RA-60 **[627]** The `collectedAddress` parameter (section 8.12.1) shall use the following format:

```
collectedAddress ::= [40]IMPLICIT AINDigits (1..24)
```

A.4.32 Encoding the numReorig parameter

RA-61 **[628]** The `numReorig` parameter (section 8.12.6) shall use the following format:

```
numReorig ::= [36] IMPLICIT INTEGER (1..255)
```

A.4.33 Encoding the jurisdictionInformation parameter

RA-62 **[629]** The `jurisdictionInformation` parameter (section 8.12.9) shall use the following format:

```
jurisdictionInformation ::= [39] IMPLICIT OCTET STRING
                                 (SIZE(3))
```

H	G	F	E	D	C	B	A
2nd Digit				1st Digit			
4th Digit				3rd Digit			
6th Digit				5th Digit			

A.4.34 Encoding the switchID parameter

RA-63 **[758]** The `switchID` parameter (section 8.12.10) shall use the following format:

```
switchID ::= [41] IMPLICIT OCTET STRING (SIZE(3))
```

The `switchID` parameter shall be 3 digits, encoded in BCD as shown here:

H	G	F	E	D	C	B	A
2nd digit				1st digit			
0	0	0	0	3rd digit			

A.4.35 Encoding the billingNumber parameter

RA-64 **[759]** The `billingNumber` parameter (section 8.12.11) shall use the following format:

```
billingNumber ::= [42]IMPLICIT AINDigits
```

A.4.36 Encoding the accountCode parameter

RA-65 **[760]** The `accountCode` parameter (section 8.12.12) shall use the following format:

```
accountCode ::= [43]IMPLICIT AINDigits
```

A.4.37 Encoding the pinDigits parameter

RA-66 **[761]** The `pinDigits` parameter (section 8.12.13) shall use the following format:

```
pinDigits ::= [44]IMPLICIT AINDigits
```

A.4.38 Encoding the connectTime parameter

RA-67 **[630]** The `connectTime` parameter (section 8.12.14) shall use the following format:

```
connectTime ::= [29] IMPLICIT OCTET STRING (SIZE(5))
```

H	G	F	E	D	C	B	A
Minutes				0	0	0	0
Minutes				Minutes			
Minutes				Minutes			
Seconds				Seconds			
0	0	0	0	Tenths			

This field is 5 octets long and is expressed in BCD. The most significant digit of the minutes will be included in octet 1, bits H-E; the second most significant digit of the minutes will be included in octet 2, bits D-A; and so on.

A.4.39 Encoding the strConnectionType parameter

RA-68 **[631]** The `strConnectionType` parameter (section 8.17.1) shall use the following format:

```
strConnectionType ::= [30] IMPLICIT ENUMERATED {
   none                          (0),
   connect_only                  (1),
   connect_1129_style            (2)
}
```

A.4.40 Encoding the pretranslatorName parameter

RA-69 **[632]** The `pretranslatorName` parameter (section 8.18.1) shall use the following format:

```
pretranslatorName ::= [31] IMPLICIT INTEGER (0..255)
```

A.4.41 Encoding the cainPRT parameter

RA-70 **[633]** The `cainPRT` parameter (section 8.19.1) shall use the following format:

```
cainPRT ::= [38] IMPLICIT AINDigits
```

RA-71 **[634]** The Nature of Address and Numbering Plan shall both be unknown.

A.4.42 Encoding the acgRequery parameter

RA-72 **[762]** The `acgRequery` parameter (section 8.20.1) shall use the following format:

```
acgRequery ::= [45] IMPLICIT NULL
-- presence = indicates the current query is an ACG
```

```
Overflow re-query
-- absence  = not a re-query
```

A.4.43 Encoding the shfelegs parameter

RA-73 **[847]** The `shfelegs` parameter (section 8.14.5) shall use the following format:

```
SHFELegs  ::=  [46] IMPLICIT OCTET STRING (SIZE(1))
```

H	G	F	E	D	C	B	A
leg7	leg6	leg5	leg4	leg3	leg2	leg1	leg0

RA-74 **[848]** Each bit will be encoded with a 1 to denote that the corresponding leg should be monitored or a 0 if that leg should not be monitored.

A.5 Encodings for extension parameters in the infoCollectedExtensionAa family

A.5.1 Encoding the accessType parameter

RA-75 [**872**] The `accessType` parameter (section 8.21.1) shall use the following format:

```
accessType ::= [5] IMPLICIT INTEGER (1)
```

Table 9-99 Values for accessType parameter

Field	numeric value
accessType	1

A.6 Encoding the FlexParameterBlock parameter

Note: The actual size of the FlexParameterBlock may be restricted by the agency in use. Refer to section 7.47 for additional information.

RA-76 **[635]** The FlexParameterBlock is encoded as follows:

```
FlexParameterBlock ::= IMPLICIT SEQUENCE (SIZE(1)) {
  IPStyle         IMPLICIT OCTET STRING (SIZE(1..115))
                                        OPTIONAL,
  SSPStyle        IMPLICIT EncodingStyle OPTIONAL
}


EncodingStyle ::= IMPLICIT SEQUENCE {
```

```
  encodingAuthority  OBJECT IDENTIFIER,
  parameters         ANY DEFINED BY encodingAuthority,
}

NTFlexParameterBlock ::= CHOICE {
  [0] VIP               VIPParameterBlock,
  [1] ServiceBuilder    ServiceBuilderFlexBlock
}

VIPParameterBlock ::= IMPLICIT SET (SIZE(1..5))
                      OF FlexCollectibleType

FlexCollectibleType ::= CHOICE {
  ANNC      [0] IMPLICIT AnncColl,
  ADDR      [1] IMPLICIT MinMaxColl,
  AUTH      [2] IMPLICIT MinMaxColl,
  PIN       [3] IMPLICIT MinMaxColl,
  ACCT      [4] IMPLICIT MinMaxColl,
  CARD      [5] IMPLICIT MinMaxColl,
  UNKNOWN   [6] IMPLICIT MinMaxColl,
  CAINPRT   [7] IMPLICIT CAINPRTColl
}


AnncColl ::= IMPLICIT OCTET STRING (SIZE(3))
```

H	G	F	E	D	C	B	A
Announcement ID (MSB)							
Announcement ID (LSB)							
0	0	0	0	0	0	0	Int

RA-77 **[636]** The interruptible flag shall be encoded as follows:

```
0     Non-interruptible
1     Interruptible


MinMaxColl ::= IMPLICIT OCTET STRING (SIZE(5))
```

H	G	F	E	D	C	B	A
Announcement ID (MSB)							
Announcement ID (LSB)							
0	0	0	MinDigs				
0	0	0	MaxDigs				
0	0	0	Int	Timer			

RA-78 **[637]** MinDigs shall be encoded as follows:

```
00000
  to        Number of digits
11000

11001
  to        Reserved Values
11111
```

RA-79 **[638]** MaxDigs shall be encoded as follows:

```
00000
  to        Number of digits
11000

11001
  to        Reserved Values
11111
```

RA-80 **[639]** Timer shall be encoded as follows:

```
0000
 to      Number of seconds

1111
```

```
CAINPrtColl ::= IMPLICIT OCTET STRING(SIZE(5))
```

H	G	F	E	D	C	B	A
Announcement ID (MSB)							
Announcement ID (LSB)							
CAINPRT (MSB)							
CAINPRT (LSB)							
0	0	0	Int	Timer			

A.7 Encoding the IPReturnBlock parameter

Note: The actual size of the IPReturnBlock may be restricted by the agency in use. Refer to section 7.36 for additional information.

RA-81 **[640]** The IPReturnBlock is encoded as follows:

```
IPReturnBlock ::= IMPLICIT SEQUENCE (SIZE (1)) {
  IPReturnBlk     IMPLICIT OCTET STRING (SIZE(1..115))
                                        OPTIONAL,
  SSPReturnBlk    IMPLICIT EncodingReturnBlock
                                        OPTIONAL
}


EncodingReturnBlock ::= IMPLICIT SEQUENCE {
  encodingAuthority     OBJECT IDENTIFIER,
```

```
  parameters            ANY DEFINED BY
                            assignmentAuthority
}


NTIPReturnBlock ::= CHOICE {
  [0] VIP               IMPLICIT SET(SIZE(1..5)) OF
                         IPCollDigits,
  [1] ServiceBuilder   ServiceBuilderIPBlock
}


IPCollDigits ::= CHOICE {
  ADDR      [1] IMPLICIT      AINDigits,
  AUTH      [2] IMPLICIT      AINDigits,
  PIN       [3] IMPLICIT      AINDigits,
  ACCT      [4] IMPLICIT      AINDigits,
  CARD      [5] IMPLICIT      AINDigits,
  UNKNOWN   [6] IMPLICIT      AINDigits,
  CAINPRT   [7] IMPLICIT      AINDigits
}
```

Appendix B: SCCP message protocol

B.1 Introduction

This appendix describes the procedures for Service Connection Control Part (SCCP) message transport of Carrier Advanced Intelligent Network (AIN) Transaction Capabilities Application Part (TCAP) messages. This appendix overrides portions of section 7.3 of GR-1299-CORE.

Global titles consist of a global title type followed by the global title number. The use of global titles allows the service switching point (SSP) to dispatch TCAP messages to service control points (SCP) via a service transfer point (STP) which determines the appropriate SCP to use based on the content and application of the message. This relieves the SSP of the knowledge of which SCPs perform what functions in the network.

For more information on SCCP messaging and global title address encoding, see GR-246-CORE, volume 2.

B.2 SCCP address procedures

The following information overrides items E and F of section 7.3.1 and section 7.3.2 of GR-2199-CORE, which describe SCCP called- and calling-party addresses for SSP initial and subsequent messages.

RB-1 **[159]** SSP initial messages shall have the called-party address encoded in global title-only form. Note, however, that UCS DMS-250 switch SCCP routing tables may be used to translate the global title prior to the query message leaving the switch, if no STP is actually in use.

RB-2 **[160]** SSP subsequent messages shall have the called-party address encoded using the SCP's address as taken from the calling-party address of the SCP's response message. This address shall contain the point code and subsystem number (PC/SSN) of the SCP's AIN application, and shall not contain a global title address. The proper return to the SCP of SSP conversational responses and error messages is dependent on a valid calling-party address.

RB-3 **[161]** The calling-party address for all SSP messages shall contain the point code and subsystem number (PC/SSN) for the AIN application on the SSP.

B.3 Global title use for AIN

In order to allow the partitioning of large AIN databases, some form of global title translation is useful, even though all AIN queries are expected to be handled by a single service, and thus a single SCCP subsystem number, at the AIN SCP.

In this release, four global title types are available. The calling line identification (CLID) global title translation type will allow convenient TCAP traffic segregation based on the caller's geographic location, as indicated by the subscriber's CLID or automatic number identification (ANI), or by the home numbering plan area (NPA) as datafilled against the originating trunk group. The address global title type will allow traffic segregation based on the called party's ID. The feature global title type will allow traffic segregation by mapping specific global title addresses to individual trigger criteria. Lastly, the office code (OFCD) global title is used exclusively for routing queries to a Local Number Portability (LNP) SCP.

With the CLID and address global title types, it is assumed that the addresses used are structured in a hierarchical fashion within a single address space, so that digit ranges may be used to break addresses into logical groups. Numbers in the national numbering plan (NANP) have this property. VPN addresses, however, may overlap depending on the customer, as well as with NANP numbers, and so do not form a single address space. For these services, the feature global title type can be used to assign an appropriate global title address to the trigger criteria that invokes each service. Note that global title addresses are typically grouped into ranges using 3 or 6 digits due to performance restrictions of the contemporary service transfer point (STP).

As specified in GR-246-CORE T1.112.3, there are two global title formats possible. The global title format is determined by the value of the global title indicator, either 0001 or 0010. Requirement R4-[265] of GR-82-CORE states that the STP only has to support the 0010 global title indicator. A global title indicator of 0010 causes an additional octet to be encoded containing the numbering plan and encoding scheme for the global title digits.

CAIN supports both the 0001 and 0010 global title indicators. In-switch provisioning indicates which encoding a particular global title uses.

RB-4 **[641]** If the global title indicator is 0001, the Encoding Scheme shall be BCD even or odd, and the Numbering Plan shall be ISDN**.**

B.3.1 CLID global title requirements

RB-5 **[162]** The NetworkBuilder implementation of AIN call processing shall provide either a 10-digit CLID or ANI from FGD and PRI agents. For DAL agents, a 3-digit NPA shall be derived from in-switch datafill. The content of the global title address shall be the same as the address transmitted in the query message's `CallingPartyID` parameter if it

has a `national` nature of address; if the CallingPartyID is an `international` address, a three-digit NPA shall be derived from in-switch datafill for use with this global title type.

RB-6 **[163]** For SSP initial messages, the called-party address indicator shall indicate that a global title in type/encoding form is included, and that the STP must perform global title translation.

RB-7 **[164v2]** The global title translation type shall be obtained from in-switch datafill for the local type name `CAIN_CLID_GT`.

B.3.2 Address global title requirements

RB-8 **[315]** AIN call processing shall provide the address digits collected from the originating agency for all agents. The content of the global title address shall be the same as the address transmitted in the query message's `CollectedAddressInfo` parameter, if present; otherwise, the address in the `CalledPartyID` parameter shall be used. If neither parameter is present, the single digit '0' shall be used.

RB-9 **[316]** For SSP initial messages, the called-party address indicator shall indicate that a global title in type/encoding form is included, that the STP must perform global title translation.

RB-10 **[317v2]** The global title translation type shall be obtained from in-switch datafill for the local type name `CAIN_ADDR_GT`.

B.3.3 Feature global title requirements

RB-11 **[443]** The content of the feature global title address shall be specified in switch datafill. The global title address digits shall be provided when the feature global title type is specified in either the `GT` option of the trigger tables or the `CAIN_DEFAULT_GT` office parameter of table `CAINPARM`.

RB-12 **[444]** For SSP initial messages, the called-party address indicator shall indicate that a global title in type/encoding form is included, and that the STP must perform global title translation.

RB-13 **[445v2]** The global title translation type shall be obtained from in-switch datafill for the local type name `CAIN_FEAT_GT`.

B.3.4 Office code global title requirements

RB-14 **[446]** The office code global title shall be used exclusively for routing queries to an LNP SCP.

RB-15 **[447v2]** For SSP initial messages, the called-party address indicator shall indicate that a global title in type/encoding form is included, and that the STP must perform global title translation.

RB-16 **[448v2]** The global title translation type shall be obtained from in-switch datafill for the local type name `CAIN_OFCD_GT`.

Appendix C: ClearCause meaning

C.1 Introduction

This appendix describes the SSP event which causes a given `ClearCause` value to be sent to the SCP in either a `Resource_Clear` or `CTR_Clear` message.

C.2 In-switch Resources

A request to connect to an in-switch resource is made in either a `Send_To_Resource` or `Connect_To_Resource` message that does not contain a `DestinationAddress` parameter.

RC-1 **[642]** Table 100 summarizes the mapping of SSP actions to ClearCauses for in-switch resources.

Table 100 SSP action to ClearCause mapping for in-switch resources (page 1 of 2)

SSP Actions	CWP or RESP package	ClearCause value
Expected result for end of dialing or announcement complete.	CWP	normal
A timeout occurred while attempting to access the resource.	CWP	timeout
The switch cancelled the resource interaction.	CWP	resourceCancelled
The caller went onhook during the resource interaction.	CWP	userAbandon
Reception of invalid digits or digit timeout.	CWP	invalidCode
The received operation could not be performed due to the unavailability of a hardware or software resource.	CWP	failure
Answer was received after an `O_No_Answer` request was sent to the SCP.	RESP	calledPartyAnswered
The switch was unable to terminate to the requested resource.	CWP	resourceNotAvailable

Table 100 SSP action to ClearCause mapping for in-switch resources (page 1 of 2)

SSP Actions	CWP or RESP package	ClearCause value
The switch determined that the request is not allowed.	CWP	taskRefused
The specified resource is not supported.	CWP	resourceTypeNotSupported
The SCP request had an encoding error and the switch can not determine a suitable action.	CWP	protocolError
The local switch encountered problems during an STR or CTR Connection.	CWP	abort
Indicates the long call duration timer expired during a STR or CTR Connection.	CWP	strCancelled

C.3 IP Resources

A request to connect to an in-switch resource is done in either a `Send_To_Resource` or `Connect_To_Resource` message that contains a `DestinationAddress` parameter.

RC-2 **[763]** If a fatal application error occurs during an active CTR connection at the Timeout R-EDP, the error is caught and a CTR_Clear message is sent instead of applying treatment. The value of the `ClearCause` parameter may not be set..

RC-3 **[643v2]** Table 9-101 summarizes the mapping of SSP actions to ClearCauses for IP resources.

Table 9-101 SSP action to ClearCause mapping for IP resources (page 1 of 3)

SSP Actions	CWP or RESP package	ClearCause value
Receipt of a `Call_Info_To_Resource` message without the `ResourceType` and `STRParameterBlock` from the SCP.	CWP	normal
The IP initiates normal call clearing with a DISCONNECT or RELEASE message with a Cause Indicator of normal and the message contains a Return Result component.		
The IP invoked a supplemental service on the local switch.	RESP	suppServiceInvoked
Termination to an IMT that is not provisioned appropriately.	CWP	taskRefused
The IPI is determined to be none.		
Cain_Conversation_Limit exceeded.		

Table 9-101 SSP action to ClearCause mapping for IP resources (page 1 of 3)

SSP Actions	CWP or RESP package	ClearCause value
The combined size of the `ResourceType` and `STRParameterBlock` has exceeded the maximum size for the given terminating agency.	CWP	abort
STR-Connection to a PRI IP and the selected trunk is not provisioned correctly.		
DISCONNECT or RELEASE message is received without a component.		
IP initiates abnormal call clearing by sending a RELEASE COMPLETE or RELEASE message with a Cause Indicator indicating abnormal clearing and the message does not contain a component.		
The DestinationAddress routes to a trunk that is not a PRI or SS7 IMT trunk.		
For the Connect Only IPI and the SSP is unable to connect to the IP at the remote SSP.		
Any other unexpected SSP errors.		
An SSP timer (not directly related to IP interactions) expires before an ISDN CONNECT or SS7 ANM message is received from the IP.	RESP	strCancelled
Local SSP is unable to locate an idle trunk member within the route list	CWP	channelsBusy
DISCONNECT or RELEASE message with a reject component is received by the SSP.	CWP	protocol error
IP initiates abnormal call clearing by sending a RELEASE COMPLETE or RELEASE message with a Cause Indicator indicating abnormal clearing and the message contains a reject component		
The IP initiates abnormal call clearing by sending a FACILITY or FAR message with a Cause Indicator indicating abnormal clearing and the message contains a reject component.		
While the SSP is waiting for a CITR from the SCP and a FACILITY or FAR containing a RO parameter with a return result component is received.		
The caller went onhook during the resource interaction.	RESP	userAbandon

Table 9-101 SSP action to ClearCause mapping for IP resources (page 1 of 3)

SSP Actions	CWP or RESP package	ClearCause value
Timer TDISC expires before the IP responds to the FACILITY or FAR message with the cancelIPresource operation.	RESP	ipTimeout
The maximum time limit for an STR-Connection has expired.	CWP	
The SCP sends a Cancel_Resource_Event message and the SSP is connected to a local IP.	CWP	resourceCancelled
The remote IP responds to a Cancel_Resource_Event with a DISCONNECT or RELEASE message containing a return result component.		
Req_Rep_BCM Non-called related component sent with a Connect_To_Resource message.	CWP	failure

RC-4 **[644]** Table 4-3 of GR-1129-CORE is used to map the Error Values in the Return Error component in the following two cases.

- The DISCONNECT or RELEASE message contains a Return Error component.
- The IP initiates abnormal call clearing and the IP sends a Facility or FAR message with a Return Error component.

Table 9-102 Summary of GR-1129-CORE Table 4-3 (page 1 of 2)

ClearCause values	Return Error/Reject
timeout	timeout
channelsBusy	channelsBusy
resourcesNotAvailable	resourcesNotAvailable
resourceTypeNotSupported	resourceTypeNotSupported
taskRefused	taskRefused
invalidCallerResponse	invalidCallerResponse
capabilityFailure	capabilityFailure
protocolError	Problem Value (Reason for component rejection)
suppServiceInvoked	suppServiceInvoked

Table 9-102 Summary of GR-1129-CORE Table 4-3 (page 1 of 2)

ClearCause values	Return Error/Reject
strCancelled	strCancelled
temporaryFailure	temporaryFailure
ipTimeout	ipTimeout
ctrCancelled	ctrCancelled

List of abbreviations

ACG
Automatic Code Gapping

AIN
Advanced Intelligent Network

ANI
automatic number identification

ASN.1
Abstract Syntax Notation One

BCD
binary-coded decimal

BCM
basic call model

CAIN
Carrier Advanced Intelligent Network (AIN)

CCID
call configuration identification

CDR
call detail record

CG
customer group

CIC
carrier identification code

CLID
calling line identification

COS
class of service

CRP
circular routing prevention

CSID
call segment identification

DAL
direct access line

DNIS
Dialed Number Inward Service

EDP
event detection point

EXTNETID
external network identifier

FGD
Feature Group D

IAM
initial address message

IEC
interexchange carrier

IN/1
Intelligent Network/1

IP
Intelligent Peripheral

IPI
Intelligent Peripheral Interface

ISDN
Integrated Services Digital Network

ISUP
Integrated Services Digital Network (ISDN) User Part

LEC
local exchange carrier

LNP
Local Number Portability

LRN
location routing number

MBG
Multi-switch Business Group

MCCS
mechanized calling card service

NCOS
Network Class of Service

NPA
numbering plan area

NEL
next event list

NETCGID
network customer group identifier

OCM
originating call model

OPART
originating partition

PANI
pseudo-automatic number identification (ANI)

PC
point code

PIC
point in call

PRI
primary rate interface

PSA
Programmable Service Architecture

RLS
release message

RRBCME
Request Report BCM Event

SAR
SCCP segmentation and reassembly

SCCP
Service Connection Control Part

SCP
service control point

SHF
switch hook flash

SLP
service logic program

SOC
software optionality control

SSN
subsystem number

SSP
service switching point

STP
service transfer point

STS
serving translation scheme

TCAP
Transaction Capabilities Application Part

TCM
terminating call model

TCN
travel card number

TDP
trigger detection point

TPART
terminating partition

UDT
unit data message

VIP
Virtual IP

XUDT
extended unit data message

Requirement-Object Index

0.1 Cross-Reference for Requirements

Absolute Reference	This Document Ref.	Page Number
1v2	R2-1	Page 4
2	R2-2	Page 4
3	R2-3	Page 4
4v3	R2-4	Page 4
5v4	R2-5	Page 5
6v2	R2-6	Page 5
7v4	R2-7	Page 6
8v4	R2-9	Page 7
9	R2-10	Page 7
10v6	R2-11	Page 9
11v4	R2-12	Page 12
12v6	R2-14	Page 13
13	R2-15	Page 24
14v4	R3-1	Page 29
15	R3-2	Page 29
16	R3-3	Page 29

17v3	R3-4	Page 30
18v4	R3-11	Page 30
19v2	R3-14	Page 31
20v3	R3-16	Page 31
21	R3-18	Page 31
22	R3-19	Page 31
23v3	R3-31	Page 33
24v4	R3-32	Page 33
25	R3-33	Page 34
26v2	R3-34	Page 34
27	R3-35	Page 34
28v2	R3-37	Page 34
29v3	R3-38	Page 34
30	R3-41	Page 34
31v2	R3-43	Page 34
32v2	R3-44	Page 34
33	R3-45	Page 34
34v2	R3-46	Page 34
35	R3-49	Page 35
36	R4-1	Page 65
37	R4-2	Page 65
38	R4-3	Page 66
39v2	R4-4	Page 67
40v5	R4-6	Page 68
41v2	R4-7	Page 68

42v3	R4-8	Page 69
43	R4-9	Page 69
44	R4-10	Page 69
45	R4-16	Page 69
46v2	R4-17	Page 69
47	R4-18	Page 70
48	R4-19	Page 70
49v3	R4-21	Page 70
50v4	R4-22	Page 70
51	R4-23	Page 70
52	R4-31	Page 71
53v4	R4-33	Page 72
54	R4-34	Page 72
55	R4-35	Page 72
56v2	R4-36	Page 72
57	R4-38	Page 72
58	R4-39	Page 72
59v4	R4-41	Page 73
60	R4-42	Page 73
61v2	R4-43	Page 73
62	R4-45	Page 73
63v2	R4-46	Page 73
64v2	R4-47	Page 73
65v5	R4-50	Page 74
66v3	R4-51	Page 74

67v2	R4-52	Page 74
68v2	R4-58	Page 75
69v3	R4-59	Page 75
70	R4-65	Page 76
71v2	R6-1	Page 99
72v4	R6-3	Page 99
73	R6-4	Page 100
74	R6-5	Page 100
75v2	R6-6	Page 100
76	R6-7	Page 100
77v2	R6-8	Page 101
78	R6-9	Page 101
79	R6-10	Page 101
80	R6-11	Page 101
81v2	R6-12	Page 101
82	R6-13	Page 101
83	R6-14	Page 101
84v2	R6-15	Page 102
85	R6-18	Page 102
86	R6-21	Page 102
87	R6-22	Page 103
88	R6-23	Page 103
89	R6-24	Page 103
90	R6-25	Page 103
91v2	R6-26	Page 103

92v2	R6-27	Page 104
93	R6-28	Page 104
94	R6-29	Page 104
95v2	R7-1	Page 105
96	R7-2	Page 105
97v2	R7-3	Page 106
98v2	R7-4	Page 106
99	R7-8	Page 107
100	R7-24	Page 110
101v3	R7-26	Page 110
102v3	R7-29	Page 111
103v2	R7-33	Page 112
104v3	R7-35	Page 113
105v4	R7-40	Page 114
106v3	R7-51	Page 117
107v3	R7-52	Page 117
108	R7-62	Page 119
109	R7-63	Page 119
110	R7-64	Page 119
111v2	R7-69	Page 120
112v2	R7-70	Page 120
113	R7-72	Page 121
114v2	R7-73	Page 121
115v2	R7-74	Page 121
116v2	R7-84	Page 123

117	R7-88	Page 124
118	R7-90	Page 124
119v3	R7-91	Page 124
20	R7-94	Page 125
121	R7-91	Page 124
122v3	R7-100	Page 127
123v4	R7-101	Page 128
124v3	R7-102	Page 128
125	R8-1	Page 131
126	R8-2	Page 131
127	R8-3	Page 131
128	R8-4	Page 131
129v2	R8-5	Page 139
130v2	R8-7	Page 139
131	R8-10	Page 139
132	R8-15	Page 140
133	R8-16	Page 140
134	R8-17	Page 140
135	R8-18	Page 140
136	R8-16	Page 140
137	R8-20	Page 140
138	R8-23	Page 141
139	R8-24	Page 141
140	R8-27	Page 141
141	R8-28	Page 141

142	R8-29	Page 141
143	R8-30	Page 141
144v4	R8-31	Page 142
145v2	R8-33	Page 142
146	RA-1	Page 191
147v4	RA-2	Page 191
148v2	RA-4	Page 193
149	RA-6	Page 193
150v4	RA-7	Page 193
151v2	RA-8	Page 194
152	RA-13	Page 196
153v2	RA-18	Page 197
154	RA-19	Page 197
155v2	RA-20	Page 197
156	RA-21	Page 197
157	RA-22	Page 198
158v2	RA-23	Page 198
159	RB-1	Page 215
160	RB-2	Page 215
161	RB-3	Page 215
162	RB-5	Page 216
163	RB-6	Page 217
164v2	RB-7	Page 217
165	R3-25	Page 32
166	R3-26	Page 32

167	R3-27	Page 32
168	R3-28	Page 32
169	R3-29	Page 32
170	R3-47	Page 35
171v2	R3-53	Page 35
172v3	R3-54	Page 36
173	R3-55	Page 36
174	R3-56	Page 36
175	R3-57	Page 36
176	R3-61	Page 37
177v2	R3-62	Page 37
178	R3-63	Page 37
179v2	R3-65	Page 37
180	R3-66	Page 37
181v3	R3-67	Page 37
182	R3-68	Page 37
183	R3-70	Page 37
184	R3-71	Page 38
185v3	R3-72	Page 38
186	R3-74	Page 39
187	R3-75	Page 39
188v3	R3-76	Page 39
189	R3-80	Page 39
190	R3-81	Page 39
191	R3-83	Page 39

192	R3-84	Page 39
193	R3-86	Page 39
194v3	R3-88	Page 40
195	R3-89	Page 40
196v3	R3-91	Page 40
197	R3-92	Page 41
198	R3-93	Page 41
199	R3-94	Page 41
200	R3-98	Page 41
201	R3-99	Page 41
202	R3-100	Page 41
203	R3-102	Page 41
204	R3-109	Page 42
205v3	R3-110	Page 43
206	R3-112	Page 43
207	R3-113	Page 44
208	R3-114	Page 44
209	R3-116	Page 44
210	R3-119	Page 44
211	R3-120	Page 44
212	R3-122	Page 44
213v2	R3-123	Page 44
214v2	R3-126	Page 44
215	R3-127	Page 44
216	R3-128	Page 45

217v3	R3-129	Page 45
218	R3-130	Page 46
219	R3-131	Page 46
220	R3-132	Page 46
221	R3-136	Page 46
222	R3-137	Page 46
223	R3-138	Page 46
224v2	R3-139	Page 46
225	R3-140	Page 46
226v2	R3-142	Page 46
227	R3-143	Page 47
228v2	R3-144	Page 47
229	R3-145	Page 47
230	R3-147	Page 47
231	R3-148	Page 47
232v3	R3-149	Page 47
233	R3-151	Page 48
234	R3-152	Page 48
235	R3-153	Page 48
236	R3-155	Page 48
237	R3-156	Page 49
238	R3-159	Page 49
239	R3-161	Page 49
240v2	R3-164	Page 49
241	R3-165	Page 49

242	R3-228	Page 57
243	R3-229	Page 57
244	R3-230	Page 57
245	R3-231	Page 57
246	R3-232	Page 57
247v3	R3-233	Page 58
248	R3-234	Page 58
249	R3-235	Page 58
250	R3-236	Page 58
251	R3-237	Page 58
252	R3-239	Page 58
253v2	R4-11	Page 69
254	R4-15	Page 69
255	R4-27	Page 70
256	R4-28	Page 71
257	R4-29	Page 71
258	R4-30	Page 71
259	R4-37	Page 72
260	R4-44	Page 73
261	R4-48	Page 74
262	R4-53	Page 75
263	R4-54	Page 75
264v2	R4-55	Page 75
265v2	R4-56	Page 75
266v4	R4-66	Page 76

267v2	R7-9	Page 107
268	R7-10	Page 108
269	R7-28	Page 111
270v2	R7-32	Page 112
271	R7-36	Page 113
272	R7-53	Page 117
273v2	R7-55	Page 118
274	R7-66	Page 119
275	R7-97	Page 125
276v2	R7-106	Page 129
277v2	R8-9	Page 139
278v3	R8-21	Page 140
279	R8-22	Page 140
280	R8-32	Page 142
281	R8-34	Page 142
282	R8-35	Page 142
283	R8-36	Page 142
284	R8-37	Page 142
285	R8-38	Page 143
286v3	R8-39	Page 143
287v3	R8-40	Page 143
288	R8-41	Page 143
289v2	R8-51	Page 144
290	R8-52	Page 144
291v2	R8-53	Page 144

292	R8-54	Page 144
293	R8-55	Page 145
294	R8-56	Page 145
295v2	R8-57	Page 145
296	R8-58	Page 145
297	R8-59	Page 145
298v3	R8-60	Page 145
299	R8-61	Page 145
300	R8-94	Page 148
301	R8-95	Page 148
302	RA-9	Page 194
303v2	RA-12	Page 196
304	RA-24	Page 198
305	RA-25	Page 198
306	RA-26	Page 199
306	RA-46	Page 203
307	RA-27	Page 199
308	RA-28	Page 199
309	RA-29	Page 199
310	RA-30	Page 199
311	RA-31	Page 199
312	RA-32	Page 200
313	RA-33	Page 200
314	RA-34	Page 200
315	RB-8	Page 217

316	RB-9	Page 217
317v2	RB-10	Page 217
318	R2-16	Page 27
319v2	R3-5	Page 30
320v2	R3-20	Page 31
321v2	R3-21	Page 31
322v3	R3-23	Page 32
323	R3-30	Page 33
324v2	R3-42	Page 34
325v2	R3-64	Page 37
326v2	R3-73	Page 38
327	R3-82	Page 39
328	R3-87	Page 40
329	R3-90	Page 40
330	R3-103	Page 42
331v2	R3-104	Page 42
332	R3-105	Page 42
333	R3-106	Page 42
334	R3-107	Page 42
335	R3-108	Page 42
336v2	R3-111	Page 43
337	R3-121	Page 44
338	R3-124	Page 44
339	R3-141	Page 46
340	R3-160	Page 49

341	R3-162	Page 49
342	R3-166	Page 49
343v2	R3-167	Page 50
344	R3-168	Page 50
345	R3-169	Page 50
346	R3-170	Page 50
347	R3-171	Page 50
348v3	R4-5	Page 67
349	R4-12	Page 69
350	R4-13	Page 69
351	R4-20	Page 70
352	R4-24	Page 70
353v2	R4-32	Page 71
354	R4-61	Page 75
355v3	R5-1	Page 89
356v4	R5-2	Page 89
357v2	R5-3	Page 90
358	R5-4	Page 90
359	R5-5	Page 90
360	R5-6	Page 90
361	R5-7	Page 90
362	R5-8	Page 90
363	R5-9	Page 90
364v2	R5-10	Page 90
365v3	R6-16	Page 102

366	R6-19	Page 102
367	R6-20	Page 102
368	R7-6	Page 107
369	R7-7	Page 107
370	R7-25	Page 110
371	R7-27	Page 110
372v2	R7-34	Page 112
373	R7-37	Page 113
374	R7-38	Page 113
375	R7-44	Page 115
376v2	R7-45	Page 115
377	R7-46	Page 116
378	R7-47	Page 116
379	R7-49	Page 116
380	R7-50	Page 117
381v2	R7-65	Page 119
382	R7-67	Page 120
383	R7-68	Page 120
384	R7-71	Page 121
385v2	R7-89	Page 124
386	R7-92	Page 125
387	R7-93	Page 125
388	R7-103	Page 129
389	R7-107	Page 129
390	R8-42	Page 143

391	R8-43	Page 143
392	R8-44	Page 143
393	R8-45	Page 143
394	R8-46	Page 143
395	R8-64	Page 145
396	R8-65	Page 145
397	R8-66	Page 146
398	R8-67	Page 146
399v2	R8-68	Page 146
400	R8-69	Page 146
401	R8-96	Page 149
402	R8-97	Page 149
403v2	R8-98	Page 149
404	R8-99	Page 149
405	R8-100	Page 149
406	R8-101	Page 149
407	R8-102	Page 149
408	R8-103	Page 149
409	R8-104	Page 149
410	R8-105	Page 149
411	R8-106	Page 149
412	R8-107	Page 149
413	R8-108	Page 150
414	R8-109	Page 150
415	R8-110	Page 150

416	R8-111	Page 150
417	R8-115	Page 150
418	R8-116	Page 150
419	R8-117	Page 151
420v2	R8-118	Page 151
421	R8-119	Page 151
422	R8-120	Page 151
423	RA-35	Page 201
424	RA-36	Page 201
425	RA-37	Page 201
426	RA-38	Page 201
427	RA-39	Page 201
428	RA-40	Page 202
429	RA-41	Page 202
430	RA-42	Page 202
431	RA-43	Page 202
432	RA-44	Page 202
433	RA-45	Page 202
434	RA-47	Page 203
435	RA-48	Page 203
436	RA-49	Page 203
437	RA-50	Page 204
438	RA-51	Page 204
439	RA-52	Page 204
440	RA-53	Page 204

441	RA-54	Page 205
442	RA-55	Page 205
443	RB-11	Page 217
444	RB-12	Page 217
445v2	RB-13	Page 217
446	RB-14	Page 217
447v2	RB-15	Page 217
448v2	RB-16	Page 218
449	R3-7	Page 30
450v2	R3-8	Page 30
451v2	R3-10	Page 30
452	R2-2	Page 4
453	R3-50	Page 35
454	R3-51	Page 35
455	R3-52	Page 35
456	R3-58	Page 36
457	R3-77	Page 39
458	R3-95	Page 41
459	R3-117	Page 44
460	R3-133	Page 46
461	R3-157	Page 49
462	R3-172	Page 50
463v2	R3-173	Page 51
464	R3-176	Page 52
465	R3-177	Page 52

466	R3-178	Page 52
467	R3-179	Page 52
468	R3-180	Page 52
469v3	R3-181	Page 52
470	R3-182	Page 52
471	R3-183	Page 52
472	R3-185	Page 52
473	R3-186	Page 52
474	R3-187	Page 52
475	R3-188	Page 52
476	R3-189	Page 53
477	R3-190	Page 53
478	R3-192	Page 53
479	R3-194	Page 53
480v3	R3-195	Page 53
481	R3-196	Page 54
482	R3-197	Page 54
483	R3-198	Page 54
484	R3-199	Page 54
485	R3-200	Page 54
486	R3-202	Page 54
487	R3-204	Page 54
488	R3-205	Page 54
489	R3-206	Page 54
490	R3-208	Page 54

491	R3-209	Page 55
492v2	R3-210	Page 55
493	R3-211	Page 55
494	R3-212	Page 55
495v2	R3-213	Page 55
496v2	R3-214	Page 55
497v2	R3-215	Page 55
498	R3-216	Page 55
499	R3-217	Page 56
500v2	R3-218	Page 56
501	R3-219	Page 56
502	R3-220	Page 56
503v2	R3-221	Page 56
504	R3-222	Page 56
505	R3-223	Page 56
506	R3-224	Page 57
507	R3-225	Page 57
508	R3-226	Page 57
509	R3-227	Page 57
510	R3-240	Page 59
511	R3-241	Page 59
512	R3-242	Page 59
513	R3-243	Page 59
514	R3-245	Page 59
515	R3-246	Page 59

516	R3-247	Page 59
517	R3-248	Page 59
518	R3-249	Page 60
519	R3-250	Page 60
520v3	R3-251	Page 60
521v2	R3-252	Page 60
522	R3-253	Page 60
523	R3-255	Page 60
524	R3-256	Page 61
525	R3-257	Page 61
526v2	R4-40	Page 72
527v3	R4-49	Page 74
528	R4-62	Page 75
529	R4-63	Page 76
530	R4-64	Page 76
531	R4-67	Page 76
532	R4-68	Page 76
533	R4-69	Page 76
534	R4-70	Page 76
535v2	R4-71	Page 76
536v3	R4-72	Page 77
537	R4-73	Page 77
538	R4-74	Page 77
539	R4-75	Page 77
540	R4-76	Page 77

541	R4-77	Page 77
542	R4-78	Page 78
543	R4-79	Page 78
544	R4-80	Page 78
545	R4-81	Page 78
546v2	R4-82	Page 78
547	R4-83	Page 79
548	R4-84	Page 79
549	R4-85	Page 79
550	R4-86	Page 79
551v2	R4-87	Page 79
552	R4-88	Page 79
553	R4-90	Page 79
554	R4-91	Page 79
555	R4-92	Page 79
556	R4-93	Page 80
557	R4-94	Page 80
558	R4-95	Page 80
559	R4-96	Page 80
560v2	R4-97	Page 80
561	R4-98	Page 80
562	R4-99	Page 80
563v2	R4-100	Page 81
564	R4-101	Page 81
565	R4-102	Page 81

566	R4-103	Page 81
567	R4-104	Page 81
568	R4-105	Page 81
569	R4-106	Page 82
570	R4-107	Page 82
571	R4-108	Page 82
572	R5-11	Page 91
573	R5-19	Page 92
574	R6-2	Page 99
575	R6-17	Page 102
576	R7-11	Page 108
577	R7-21	Page 109
578	R7-30	Page 111
579	R7-31	Page 112
580v2	R7-39	Page 113
581v2	R7-48	Page 116
582	R7-56	Page 118
583	R7-57	Page 118
584	R7-58	Page 118
585	R7-59	Page 118
586v2	R7-60	Page 118
587v2	R7-61	Page 119
588	R7-75	Page 121
589	R7-76	Page 121
590	R7-77	Page 122

591	R7-78	Page 122
592	R7-79	Page 122
593v2	R7-83	Page 123
594	R7-95	Page 125
595	R7-98	Page 125
596v2	R7-99	Page 126
597	R8-6	Page 139
598	R8-8	Page 139
599	R8-11	Page 139
600	R8-12	Page 139
601	R8-13	Page 139
602	R8-14	Page 139
603	R8-25	Page 141
604	R8-26	Page 141
605v2	R8-47	Page 144
606	R8-48	Page 144
607v2	R8-49	Page 144
608	R8-50	Page 144
609	R8-62	Page 145
610	R8-63	Page 145
611v2	R8-70	Page 146
612	R8-71	Page 146
613	R8-92	Page 148
614	R8-93	Page 148
615	R8-121	Page 151

616	R8-122	Page 151
617	R8-125	Page 152
618	R8-126	Page 152
619	RA-14	Page 196
620	RA-15	Page 197
621	RA-16	Page 197
622	RA-17	Page 197
623	RA-56	Page 205
624	RA-57	Page 205
625	RA-58	Page 205
626v2	RA-59	Page 206
627	RA-60	Page 206
628	RA-61	Page 206
629	RA-62	Page 206
630	RA-67	Page 207
631	RA-68	Page 208
632	RA-69	Page 208
633	RA-70	Page 208
634	RA-71	Page 208
635	RA-76	Page 209
636	RA-77	Page 210
637	RA-78	Page 210
638	RA-79	Page 211
639	RA-80	Page 211
640	RA-81	Page 211

641	RB-4	Page 216
642	RC-1	Page 219
643v2	RC-3	Page 220
644	RC-4	Page 222
645	R2-8	Page 6
646	R3-12	Page 31
647	R3-13	Page 31
648	R3-15	Page 31
649	R3-17	Page 31
650	R3-48	Page 35
651	R3-69	Page 37
652	R3-85	Page 39
653	R3-101	Page 41
654	R3-125	Page 44
655	R3-146	Page 47
656	R3-163	Page 49
657	R3-191	Page 53
658	R3-207	Page 54
659	R3-238	Page 58
660	R3-254	Page 60
661	R4-57	Page 75
662	R4-89	Page 79
663	R5-12	Page 91
664	R5-13	Page 91
665	R5-14	Page 91

666	R5-15	Page 91
667	R5-16	Page 91
668	R5-17	Page 91
669	R5-18	Page 91
670	R5-20	Page 92
671	R5-21	Page 92
672	R5-22	Page 92
673	R5-23	Page 92
674	R5-24	Page 92
675	R5-25	Page 92
676	R5-26	Page 92
677	R5-27	Page 92
678	R5-28	Page 92
679	R5-29	Page 92
680	R5-30	Page 93
681	R5-31	Page 93
682	R5-32	Page 93
683	R5-33	Page 93
684	R5-34	Page 93
685	R5-35	Page 93
686	R5-36	Page 93
687	R5-42	Page 94
688	R5-43	Page 95
689	R5-44	Page 95
690	R5-45	Page 95

691	R5-46	Page 95
692	R5-47	Page 95
693	R5-48	Page 95
694	R5-49	Page 95
695	R5-50	Page 96
696	R5-51	Page 96
697	R5-52	Page 96
698	R5-53	Page 96
699	R5-54	Page 96
700	R5-55	Page 96
701	R5-56	Page 96
702	R5-57	Page 96
703	R5-58	Page 96
704	R5-59	Page 96
705	R5-60	Page 96
706	R5-61	Page 97
707	R5-62	Page 97
708	R5-63	Page 97
709	R5-64	Page 97
710	R5-65	Page 97
711	R5-66	Page 97
712	R5-67	Page 97
713	R5-68	Page 97
714	R5-69	Page 98
715	R5-70	Page 98

716	R5-71	Page 98
721	R7-5	Page 106
722	R7-22	Page 110
723	R7-23	Page 110
725v2	R7-41	Page 114
726v2	R7-42	Page 115
727	R7-85	Page 124
728	R7-86	Page 124
729	R7-87	Page 124
730v2	R7-104	Page 129
731	R8-72	Page 146
732	R8-73	Page 146
733	R8-74	Page 146
734	R8-75	Page 146
735	R8-76	Page 147
736	R8-77	Page 147
737	R8-78	Page 147
738	R8-79	Page 147
739	R8-80	Page 147
740	R8-81	Page 147
741	R8-82	Page 147
742	R8-83	Page 147
743	R8-84	Page 147
744	R8-85	Page 147
745	R8-86	Page 147

746	R8-87	Page 147
747	R8-88	Page 147
748	R8-89	Page 148
749	R8-90	Page 148
750	R8-91	Page 148
751	R8-123	Page 151
752	R8-124	Page 151
753	R8-127	Page 152
754	R8-128	Page 152
755	R8-129	Page 152
756	RA-10	Page 195
757	RA-11	Page 195
758	RA-63	Page 207
759	RA-64	Page 207
760	RA-65	Page 207
761	RA-66	Page 207
762	RA-72	Page 208
763	RC-2	Page 220
764v3	R2-13	Page 13
765	R3-40	Page 34
766v2	R3-59	Page 36
767	R3-78	Page 39
768	R3-96	Page 41
769	R3-118	Page 44
770	R3-134	Page 46

771	R3-158	Page 49
772	R3-184	Page 52
773	R3-201	Page 54
774	R4-111	Page 82
775	R4-128	Page 85
776	R4-131	Page 86
777	R3-193	Page 53
778	R4-138	Page 87
779	R3-258	Page 61
780	R3-259	Page 61
781	R3-260	Page 61
782	R3-261	Page 61
783	R3-262	Page 61
784	R3-263	Page 62
785	R3-264	Page 62
786	R3-265	Page 62
787	R4-110	Page 82
788	R4-112	Page 83
789	R4-113	Page 83
790	R4-114	Page 83
791	R4-115	Page 84
792	R4-116	Page 84
793	R4-117	Page 84
794	R4-118	Page 84
795	R4-119	Page 84

796	R4-120	Page 84
797	R4-121	Page 84
798	R4-122	Page 84
799	R4-123	Page 85
800	R4-124	Page 85
801	R4-125	Page 85
802	R4-126	Page 85
803	R4-127	Page 85
804	R4-129	Page 85
805	R4-130	Page 86
806	R4-132	Page 86
807	R4-133	Page 86
808	R4-134	Page 86
809	R4-135	Page 86
810	R4-139	Page 87
811	R4-140	Page 87
812	R4-141	Page 87
813	R3-266	Page 62
814	R3-22	Page 32
815	R3-24	Page 32
816	R3-36	Page 34
817	R3-60	Page 36
818	R3-79	Page 39
819	R3-97	Page 41
820	R3-115	Page 44

821	R3-135	Page 46
822	R3-154	Page 48
823	R3-203	Page 54
824	R7-12	Page 108
825	R7-13	Page 108
826	R7-14	Page 109
827	R7-15	Page 109
828	R7-16	Page 109
829	R7-17	Page 109
830	R7-18	Page 109
831	R7-19	Page 109
832	R7-20	Page 109
833	R7-80	Page 122
834	R7-81	Page 122
835	R7-82	Page 122
837	R3-267	Page 62
838	R3-268	Page 62
839	R3-269	Page 62
840	R3-270	Page 63
841	R3-271	Page 63
842	R3-272	Page 63
843	R7-54	Page 117
844	R8-112	Page 150
845	R8-113	Page 150
846	R8-114	Page 150

847	RA-73	Page 209
848	RA-74	Page 209
849	R4-137	Page 86
850	R4-136	Page 86
851	R4-25	Page 70
852	R4-26	Page 70
853	R4-109	Page 82
854	R3-6	Page 30
855	R3-9	Page 30
856	R3-150	Page 48
857	R4-14	Page 69
858	R4-60	Page 75
859	R5-37	Page 94
860	R5-38	Page 94
861	R5-39	Page 94
862	R5-40	Page 94
863	R5-41	Page 94
864	R7-105	Page 129
865	RA-3	Page 192
866	R7-43	Page 115
867	R8-130	Page 152
868	R8-131	Page 152
869	R8-132	Page 152
870	R8-133	Page 153
871	RA-5	Page 193

872 RA-75 Page 209

UCS DMS-250

NetworkBuilder Carrier AIN

TCAP Protocol Definition

Product Documentation — Dept 3423
Nortel Networks
P.O. Box 13010
RTP, NC 27709-3010
1-800-684-2273
(1-800-NTI-CARE)

PROPRIETARY
Released under license

NORTEL NETWORKS PROPRIETARY: The information contained in this document is the property of Nortel Networks. Except as specifically authorized in writing by Nortel Networks, the holder of this document shall keep the information contained herein confidential and shall protect same in whole or in part from disclosure and dissemination to third parties and use same for evaluation, operation, and maintenance purposes only. Information is subject to change without notice.

Nortel Networks reserves the right to make changes in design or components as progress in engineering and manufacturing may warrant

DMS, MAP, NetworkBuilder, Nortel, and SuperNode are trademarks of Nortel Networks.

Publication number: NIS D354-1
Product release: UCS15 (CSP15)
Document release: Release 08.01
Date: June 2001
Printed in the United States of America