

# Mediant™ 2000 VoIP Media Gateway

## Mediant™ 2000 & TP-1610 SIP User's Manual

Version 4.4

Document #: LTRT-72504



## Notice

This document describes the AudioCodes Mediant™ 2000 SIP (Session Initialization Protocol) gateway and the TP-1610 SIP cPCI board.

Information contained in this document is believed to be accurate and reliable at the time of printing. However, due to ongoing product improvements and revisions, AudioCodes cannot guarantee accuracy of printed material after the Date Published nor can it accept responsibility for errors or omissions. Updates to this document and other documents can be viewed by registered Technical Support customers at [www.audiocodes.com](http://www.audiocodes.com) under Support / Product Documentation.

© Copyright 2005 AudioCodes Ltd. All rights reserved.

This document is subject to change without notice.

Date Published: Jul-18-2005

Date Printed: Jul-19-2005

---

## Table of Contents

---

<b>1</b>	<b>Overview</b>	<b>13</b>
1.1	Available Configurations	14
1.2	SIP Overview	15
1.3	Mediant 2000 Features	15
1.3.1	General Features	15
1.3.2	Hardware Features	15
1.3.3	PSTN-to-SIP Interworking	16
1.3.3.1	Supported Interworking Features	16
1.3.4	Supported SIP Features	16
<b>2</b>	<b>Mediant 2000 Physical Description</b>	<b>19</b>
2.1	General	19
2.2	The Mediant 2000 Chassis	20
2.2.1	Power Supply	20
2.3	The TP-1610 Board	20
2.3.1	Board Hot-Swap Support	21
2.3.1.1	Removing Boards	22
2.3.1.2	Inserting Boards	22
2.3.2	TP-1610 Front Panel LED Indicators	23
2.4	Rear Transition Module	24
2.5	Optional CPU Board	25
<b>3</b>	<b>Installing the Mediant 2000</b>	<b>27</b>
3.1	Unpacking	27
3.2	Package Contents	27
3.3	Mounting the Mediant 2000	28
3.3.1	Mounting the Mediant 2000 on a Desktop	28
3.3.2	Installing the Mediant 2000 in a 19-inch Rack	28
3.4	Cabling the Mediant 2000	30
3.4.1	Connecting the E1/T1 Trunk Interfaces	31
3.4.2	Installing the Ethernet Connection	32
3.4.3	Connecting the Power Supply	33
3.4.3.1	Connecting the AC Power Supply	33
3.4.3.2	Connecting the DC Power Supply	33
<b>4</b>	<b>Getting Started</b>	<b>35</b>
4.1	Assigning the Mediant 2000 IP Address	35
4.1.1	Assigning an IP Address Using HTTP	35
4.1.2	Assigning an IP Address Using BootP	36
4.2	Restoring Networking Parameters to their Initial State	36
4.3	Configuring the Mediant 2000 <i>Basic</i> Parameters	37
<b>5</b>	<b>Web Management</b>	<b>39</b>
5.1	Configuration Concepts	39
5.2	Overview of the Embedded Web Server	39
5.3	Computer Requirements	39
5.4	Password Control	40
5.4.1	Embedded Web Server Username & Password	40
5.5	Configuring the Web Interface via the <i>ini</i> File	40
5.5.1	Limiting the Embedded Web Server to Read-Only Mode	40
5.5.2	Disabling the Embedded Web Server	40
5.6	Accessing the Embedded Web Server	41
5.6.1	Using Internet Explorer to Access the Embedded Web Server	41
5.7	Getting Acquainted with the Web Interface	42
5.7.1	Main Menu Bar	42
5.7.2	Saving Changes	43
5.7.3	Entering Phone Numbers in Various Tables	43
5.8	Protocol Management	44
5.8.1	Protocol Definition Parameters	44
5.8.1.1	Coders	44
5.8.2	Advanced Parameters	45
5.8.3	Number Manipulation Tables	45
5.8.3.1	Dialing Plan Notation	47

5.8.3.2	Numbering Plans and Type of Number .....	48
5.8.4	Configuring the Routing Tables .....	49
5.8.4.1	Tel to IP Routing Table .....	49
5.8.4.2	IP to Trunk Group Routing Table .....	51
5.8.4.3	Internal DNS Table .....	53
5.8.4.4	Reasons for Alternative Routing .....	54
5.8.5	Configuring the Profile Definitions .....	55
5.8.5.1	Coder Group Settings .....	55
5.8.5.2	Tel Profile Settings .....	56
5.8.5.3	IP Profile Settings .....	57
5.8.6	Configuring the Trunk Group Table .....	58
5.8.7	Configuring the Trunk Group Settings .....	60
5.9	Advanced Configuration .....	62
5.9.1	Configuring the Network Settings .....	62
5.9.1.1	Configuring the SNMP Managers Table .....	63
5.9.1.2	Multiple Routers Support .....	63
5.9.1.3	Simple Network Time Protocol Support .....	63
5.9.2	Configuring the Channel Settings .....	65
5.9.3	Configuring the Trunk Settings .....	66
5.9.4	Configuring the TDM Bus Settings .....	68
5.9.5	Restoring and Backing up the Gateway Configuration .....	69
5.9.6	Regional Settings .....	70
5.9.7	Changing the Mediant 2000 Username and Password .....	71
5.10	Status & Diagnostic .....	71
5.10.1	Gateway Statistics .....	71
5.10.1.1	IP Connectivity .....	71
5.10.1.2	Call Counters .....	73
5.10.2	Monitoring the Mediant 2000 Trunks & Channels .....	75
5.10.3	Activating the Internal Syslog Viewer .....	76
5.10.4	System Information .....	77
5.11	Software Update Menu .....	78
5.11.1	Software Upgrade Wizard .....	78
5.11.2	Auxiliary Files .....	82
5.11.3	Updating the Software Upgrade Key .....	83
5.12	Save Configuration .....	84
5.13	Resetting the Mediant 2000 .....	85
<b>6</b>	<b><i>ini</i> File Configuration of the Mediant 2000 .....</b>	<b>87</b>
6.1	Secured <i>ini</i> File .....	87
6.2	Modifying an <i>ini</i> File .....	87
6.3	The <i>ini</i> File Content .....	88
6.4	The <i>ini</i> File Structure .....	88
6.4.1	The <i>ini</i> File Structure Rules .....	88
6.5	The <i>ini</i> File Example .....	89
6.6	Basic, Logging, Web and RADIUS Parameters .....	90
6.7	SNMP Parameters .....	98
6.8	SIP Configuration Parameters .....	100
6.9	ISDN and CAS Interworking-Related Parameters .....	111
6.10	Number Manipulation and Routing Parameters .....	115
6.11	E1/T1 Configuration Parameters .....	122
6.12	Channel Parameters .....	128
6.12.1	Dynamic Jitter Buffer Operation .....	132
6.13	Configuration Files Parameters .....	133
<b>7</b>	<b>Configuration Files .....</b>	<b>135</b>
7.1	Configuring the Call Progress Tones .....	135
7.1.1	Format of the Call Progress Tones Section in the <i>ini</i> File .....	135
7.2	Pre-recorded Tones (PRT) File .....	137
7.2.1	PRT File Format .....	137
7.3	Voice Prompts File .....	137
7.4	CAS Protocol Configuration Files .....	138
<b>8</b>	<b>Gateway Capabilities Description .....</b>	<b>139</b>
8.1	Proxy or Registrar Registration Example .....	139
8.2	Redirect Number and Calling Name (Display) .....	139
8.3	ISDN Overlap Dialing .....	140

8.4	Using ISDN NFAS .....	141
8.4.1	NFAS Interface ID .....	141
8.4.2	Working with DMS-100 Switches .....	142
8.5	Configuring the DTMF Transport Types .....	143
8.6	Configuring the Gateway's Alternative Routing (based on Connectivity and QoS).....	146
8.6.1	Alternative Routing Mechanism.....	146
8.6.2	Determining the Availability of Destination IP Addresses .....	146
8.6.3	PSTN Fallback as a Special Case of Alternative Routing.....	146
8.6.4	Relevant Parameters.....	147
8.7	Working with Supplementary Services .....	147
8.7.1	Call Hold and Retrieve Features .....	147
8.7.2	Call Transfer.....	147
8.8	TDM Tunneling .....	149
8.8.1	Implementation.....	149
8.9	Call Detail Report.....	151
8.10	Trunk to Trunk Routing Example.....	152
8.11	SIP Call Flow Example .....	153
8.12	SIP Authentication Example .....	156
8.13	Nortel IMS SIP2PRI Gateway Specific Features and Configuration .....	158
8.13.1	SIP to PRI Calls.....	158
8.13.2	PRI to SIP Calls.....	159
8.13.3	Support for RPI Header.....	160
8.13.3.1	Configuration of NPI/TON .....	160
8.13.4	Transfer .....	161
8.13.5	Other Nortel Specific Parameters.....	161
8.14	Nortel IMS SIP2CAS (Call Pilot) Gateway Specific Features and Configuration.....	162
8.14.1	Supported Features.....	162
8.15	DTMF Configuration for Nortel Gateways .....	163
<b>9</b>	<b>Diagnostics .....</b>	<b>165</b>
9.1	Mediant 2000 Self-Testing.....	165
9.2	Syslog Support .....	165
9.2.1	Syslog Servers .....	166
9.2.2	Operation.....	166
9.2.2.1	Sending the Syslog Messages.....	166
9.2.2.2	Setting the Syslog Server.....	166
9.2.2.3	The <i>ini</i> File Example for Syslog.....	166
<b>10</b>	<b>BootP/DHCP Support .....</b>	<b>167</b>
10.1	Startup Process .....	167
10.2	DHCP Support.....	169
10.3	BootP Support .....	169
10.3.1	Upgrading the Mediant 2000 .....	169
10.3.2	Vendor Specific Information Field .....	170
<b>11</b>	<b>SNMP-Based Management.....</b>	<b>171</b>
11.1	About SNMP .....	171
11.1.1	SNMP Message Standard.....	171
11.1.2	SNMP MIB Objects .....	172
11.1.3	SNMP Extensibility Feature.....	172
11.2	Carrier Grade Alarm System .....	173
11.2.1	Active Alarm Table .....	173
11.2.2	Alarm History .....	173
11.3	Cold Start Trap .....	173
11.4	Third-Party Performance Monitoring Measurements .....	174
11.5	TrunkPack-VoP Series Supported MIBs .....	174
11.6	SNMP Interface Details .....	177
11.6.1	SNMP Community Names .....	177
11.6.1.1	Configuration of Community Strings via the <i>ini</i> File.....	177
11.6.1.2	Configuration of Community Strings via SNMP.....	177
11.6.2	Trusted Managers .....	178
11.6.2.1	Configuration of Trusted Managers via <i>ini</i> File.....	178
11.6.2.2	Configuration of Trusted Managers via SNMP.....	178
11.6.3	SNMP Ports.....	179
11.6.4	Multiple SNMP Trap Destinations .....	180
11.6.4.1	Configuration via the <i>ini</i> File.....	180

11.6.4.2 Configuration via SNMP .....	181
11.7 SNMP Manager Backward Compatibility.....	182
11.8 AudioCodes' Element Management System.....	182
<b>12 Selected Technical Specifications .....</b>	<b>183</b>
<b>Appendix A Mediant 2000 SIP Software Kit.....</b>	<b>187</b>
<b>Appendix B The BootP/TFTP Configuration Utility .....</b>	<b>189</b>
B.1 When to Use the BootP/TFTP .....	189
B.2 An Overview of BootP.....	189
B.3 Key Features .....	189
B.4 Specifications.....	190
B.5 Installation.....	190
B.6 Loading the <i>cmp</i> File, Booting the Device .....	190
B.7 BootP/TFTP Application User Interface.....	191
B.8 Function Buttons on the Main Screen .....	191
B.9 Log Window .....	192
B.10 Setting the Preferences .....	193
B.10.1 BootP Preferences .....	193
B.10.2 TFTP Preferences .....	194
B.11 Configuring the BootP Clients.....	195
B.11.1 Adding Clients .....	195
B.11.2 Deleting Clients .....	196
B.11.3 Editing Client Parameters.....	196
B.11.4 Testing the Client .....	196
B.11.5 Setting Client Parameters .....	197
B.11.6 Using Command Line Switches .....	198
B.12 Managing Client Templates.....	199
<b>Appendix C RTP/RTCP Payload Types and Port Allocation .....</b>	<b>201</b>
C.1 Payload Types Defined in RFC 1890 .....	201
C.2 Defined Payload Types.....	201
C.3 Default RTP/RTCP/T.38 Port Allocation.....	202
<b>Appendix D Fax and Modem Transport Modes.....</b>	<b>203</b>
D.1 Fax/Modem Settings.....	203
D.1.1 Configuring Fax Relay Mode.....	203
D.1.2 Configuring Fax/Modem ByPass Mode.....	203
D.1.3 Supporting V.34 Faxes.....	204
<b>Appendix E Mediant 2000 Clock Settings .....</b>	<b>205</b>
<b>Appendix F Customizing the Mediant 2000 Web Interface .....</b>	<b>207</b>
F.1 Replacing the Main Corporate Logo.....	207
F.1.1 Replacing the Main Corporate Logo with an Image File.....	207
F.1.2 Replacing the Main Corporate Logo with a Text String.....	209
F.2 Replacing the Background Image File.....	209
F.3 Customizing the Product Name.....	210
F.4 Modifying <i>ini</i> File Parameters via the Web AdminPage .....	211
<b>Appendix G Accessory Programs and Tools.....</b>	<b>213</b>
G.1 TrunkPack Downloadable Conversion Utility.....	213
G.1.1 Converting a CPT <i>ini</i> File to a Binary <i>dat</i> File .....	214
G.1.2 Creating a Loadable Voice Prompts File.....	215
G.1.3 Encoding / Decoding an <i>ini</i> File.....	217
G.1.4 Creating a Loadable Pre-recorded Tones File .....	218
G.2 PSTN Trace Utility .....	220
G.2.1 Operation.....	220
<b>Appendix H Software Upgrade Key .....</b>	<b>223</b>
H.1 About the Software Upgrade Key .....	223
H.2 Backing up the Current Software Upgrade Key.....	223
H.3 Loading the Software Upgrade Key.....	223
H.3.1 Loading the Software Upgrade Key Using the Embedded Web Server .....	224
H.3.2 Loading the Software Upgrade Key Using BootP/TFTP .....	225
H.4 Verifying that the Key was Successfully Loaded .....	225
H.5 Troubleshooting an Unsuccessful Loading of a Key .....	225
H.6 Abort Procedure.....	225
<b>Appendix I Release Reason Mapping.....</b>	<b>227</b>

<b>Appendix J SS7 Tunneling</b> .....	<b>231</b>
J.1 MTP2 Tunneling Technology.....	232
J.2 SS7 Characteristics .....	232
J.3 SS7 Parameters .....	233
J.4 SS7 Table Parameters .....	234
J.4.1 SIGTRAN Interface Groups.....	234
J.4.2 SIGTRAN Interface IDs .....	235
J.4.3 SS7 Signaling Link .....	236
J.5 SS7 MTP2 Tunneling <i>ini</i> File Example .....	237
J.6 <i>ini</i> File Parameters in a Table Format .....	241
J.6.1 Table Indices .....	242
J.6.2 Table Permissions .....	242
J.6.3 Tables of Parameter Value Rules in the <i>ini</i> File Structure .....	243
J.6.3.1 Tables Structure Rules.....	243
J.6.3.2 Dynamic Tables versus Static Tables .....	244
J.6.3.3 Tables in the Loaded <i>ini</i> File .....	244
<b>Appendix K RADIUS Billing and VXML Calling Card Application</b> .....	<b>245</b>
K.1 Benefits.....	245
K.2 Features.....	245
K.3 Supported Architecture .....	246
K.4 Implementation .....	247
K.4.1 Basic Calling Card IVR Scenario.....	247
K.4.2 Call Flow Description.....	248
K.5 Operation & Configuration .....	249
K.6 Configuration Parameters.....	249
K.7 Supported RADIUS Attributes .....	251
K.8 RADIUS Server Messages .....	253
K.8.1 Authentication.....	253
12.1.1 Authorization.....	253
12.1.2 Accounting.....	254
K.9 Voice XML Interpreter.....	254
K.9.1 Features .....	254
K.10 Supported Elements & Attributes .....	256
K.11 Provided Calling Card System.....	260
K.11.1 Voice Prompts .....	260
K.11.2 VXML Flow Chart .....	262
K.12 VXML Script Example.....	266
<b>Appendix L SNMP Traps</b> .....	<b>271</b>
L.1 Alarm Traps .....	271
L.1.1 Component: System#0.....	271
L.1.2 Component: AlarmManager#0 .....	275
L.1.3 Component: EthernetLink#0.....	275
L.1.4 Other Traps .....	276
L.1.5 Trap Varbinds .....	276
<b>Appendix M Regulatory Information</b> .....	<b>277</b>

## List of Figures

Figure 1-1: Typical Mediant 2000 Gateway Application .....	14
Figure 2-1: Mediant 2000 Front View .....	19
Figure 2-2: Front and Upper View of the TP-1610 cPCI Board .....	21
Figure 2-3: Rear Panel with two 50-pin Connectors for 16 Trunks .....	24
Figure 2-4: Rear Panel with 8 RJ-48c Connectors for 8 Trunks .....	25
Figure 3-1: 19-inch Rack & Desktop Accessories .....	28
Figure 3-2: Mediant 2000 Front View with 19-inch Rack Mount Brackets .....	29
Figure 3-3: Mediant 2000 Rear Panel Cabling (16 Trunks, Dual AC Power) .....	30
Figure 3-4: Mediant 2000 Rear Panel Cabling (8 Trunks, DC Power) .....	31
Figure 3-5: 50-pin Female Telco Board-Mounted Connector .....	32
Figure 3-6: Pinout of RJ-48c Trunk Connectors .....	32
Figure 3-7: Pinout of RJ-45 Connectors .....	33
Figure 3-8: DC Terminal Block Screw Connector .....	34
Figure 3-9: DC Terminal Block Crimp Connector .....	34
Figure 4-1: Mediant 2000 Quick Setup Screen .....	37
Figure 5-1: Embedded Web Server Login Screen .....	41
Figure 5-2: Mediant 2000 Web Interface .....	42
Figure 5-3: Coders Screen .....	44
Figure 5-4: Source Phone Number Manipulation Table for Tel→IP Calls .....	46
Figure 5-5: Tel to IP Routing Table Screen .....	50
Figure 5-6: IP to Trunk Group Routing Table .....	52
Figure 5-7: Internal DNS Table Screen .....	53
Figure 5-8: Reasons for Alternative Routing Screen .....	54
Figure 5-9: Coder Group Settings Screen .....	55
Figure 5-10: Tel Profile Settings Screen .....	56
Figure 5-11: IP Profile Settings Screen .....	57
Figure 5-12: Trunk Group Table Screen .....	58
Figure 5-13: Trunk Group Settings Screen .....	60
Figure 5-14: Network Settings Screen .....	62
Figure 5-15: SNMP Managers Table Screen .....	63
Figure 5-16: Channel Settings Screen .....	65
Figure 5-17: E1/T1 Trunk Settings Screen .....	66
Figure 5-18: TDM Bus Settings Screen .....	68
Figure 5-19: Configuration File Screen .....	69
Figure 5-20: Regional Settings Screen .....	70
Figure 5-21: Change Password Screen .....	71
Figure 5-22: IP Connectivity Screen .....	72
Figure 5-23: Tel→IP Call Counters Screen .....	73
Figure 5-24: Mediant 2000 Trunk & Channel Status Screen .....	75
Figure 5-25: Trunk and Channel Status Color Indicator Keys .....	75
Figure 5-26: Channel Status Details Screen .....	76
Figure 5-27: Message Log Screen .....	76
Figure 5-28: System Information Screen .....	77
Figure 5-29: Start Software Upgrade Screen .....	78
Figure 5-30: Load a <i>cmp</i> File Screen .....	79
Figure 5-31: <i>cmp</i> File Successfully Loaded into the Mediant 2000 Notification .....	79
Figure 5-32: Load an <i>ini</i> File Screen .....	80
Figure 5-33: Load a CPT File Screen .....	81
Figure 5-34: FINISH Screen .....	81
Figure 5-35: 'End Process' Screen .....	82
Figure 5-36: Auxiliary Files Screen .....	83
Figure 5-37: Save Configuration Screen .....	84
Figure 5-38: Reset Screen .....	85
Figure 6-1: <i>ini</i> File Structure .....	88
Figure 6-2: SIP <i>ini</i> File Example .....	89
Figure 7-1: Call Progress Tone Types .....	136
Figure 7-2: Defining a Dial Tone Example .....	136
Figure 8-1: <i>ini</i> File Example for TDM Tunneling (Originating Side) .....	150
Figure 8-2: <i>ini</i> File Example for TDM Tunneling (Terminating Side) .....	150

Figure 8-3: SIP Call Flow Example.....	153
Figure 5-2: IP to Trunk Group Routing Table.....	159
Figure 9-1: Setting the Syslog Server IP Address.....	166
Figure 9-2: The <i>ini</i> File Example for Syslog.....	166
Figure 10-1: Mediant 2000 Startup Process.....	168
Figure 11-1: Example of Entries in a Device <i>ini</i> file Regarding SNMP.....	181
Figure B-1: Main Screen.....	191
Figure B-2: Reset Screen.....	191
Figure B-3: Preferences Screen.....	193
Figure B-4: Client Configuration Screen.....	195
Figure B-5: Templates Screen.....	199
Figure F-1: User-Customizable Web Interface Title Bar.....	207
Figure F-2: Customized Web Interface Title Bar.....	207
Figure F-3: Image Download Screen.....	208
Figure F-4: INI Parameters Screen.....	211
Figure G-1: TrunkPack Downloadable Conversion Utility Opening Screen.....	213
Figure G-2: Call Progress Tones Conversion Screen.....	214
Figure G-3: Voice Prompts Screen.....	215
Figure G-4: File Data Window.....	216
Figure G-5: Encode/Decode <i>ini</i> File(s) Screen.....	217
Figure G-6: Prerecorded Tones Screen.....	218
Figure G-7: File Data Window.....	219
Figure H-8: Trunk Traces.....	221
Figure H-9: UDP2File Utility.....	221
Figure H-1: Software Upgrade Key Screen.....	224
Figure H-2: Example of a Software Upgrade Key File Containing Multiple S/N Lines.....	225
Figure J-1: M2UA Architecture.....	231
Figure J-2: M2TN Architecture.....	231
Figure J-3: Protocol Architecture for MTP2 Tunneling.....	232
Figure J-4: SS7 MTP2 Tunneling <i>ini</i> File Example - MGC.....	238
Figure J-5: SS7 MTP2 Tunneling <i>ini</i> File Example - SG.....	240
Figure J-6: Structure of a Table in an <i>ini</i> File.....	243
Figure K-1: Mediant 2000 Supported Architecture.....	246
Figure K-2: Basic Call Scenario.....	247
Figure K-3: Basic <i>ini</i> File VXML Parameters.....	248
Figure K-4: Authentication Example.....	253
Figure K-5: Authorization Example.....	253
Figure K-6: Accounting Example.....	254
Figure K-7: VXML Script Opening Menu.....	262
Figure K-8: VXML Script Option 1, Make a Call.....	263
Figure K-9: VXML Script, Call Transfer Procedure.....	264
Figure K-10: VXML Script, Options 2, 3 and 4.....	265
Figure K-11: VXML Script, Call Termination.....	265
Figure K-12: VXML Script Example (continues on pages 261 to 265).....	266

## List of Tables

Table 2-1: Mediant 2000 Front View Component Descriptions.....	19
Table 2-2: Chassis LED Indicators.....	20
Table 2-3: Front and Upper View of the TP-1610 cPCI Board Component Descriptions.....	21
Table 2-4: Status LED Indicators.....	23
Table 2-5: E1/T1 Trunk Status LED Indicators.....	23
Table 2-6: Ethernet LED Indicators.....	23
Table 2-7: cPCI LED Indicators.....	23
Table 2-8: Rear Panel with two 50-pin Connectors for 16 Trunks Component Descriptions.....	24
Table 2-9: Rear Panel with 8 RJ-48c Connectors for 8 Trunks Component Descriptions.....	25
Table 3-1: Mediant 2000 Rear Panel Cabling (16 Trunks, Dual AC Power) Component Descriptions.....	30
Table 3-2: Mediant 2000 Rear Panel Cabling (8 Trunks, DC Power) Component Descriptions.....	31
Table 3-3: E1/T1 Connections on each 50-pin Telco Connector.....	32
Table 4-1: Mediant 2000 Default Networking Parameters.....	35
Table 5-1: Number Manipulation Parameters.....	46
Table 5-2: NPI/TON Values for ISDN ETSI.....	48
Table 5-3: Tel to IP Routing Table.....	50
Table 5-4: IP to Trunk Group Routing Table.....	52
Table 5-5: Trunk Group Table.....	59
Table 5-6: Channel Select Modes.....	61
Table 5-7: Trunks Status Color Indicator Keys.....	67
Table 5-8: IP Connectivity Parameters.....	72
Table 5-9: Call Counters Description (continues on pages 73 to 74).....	73
Table 5-10: Auxiliary Files Descriptions.....	82
Table 6-1: Basic, Logging, Web and RADIUS Parameters (continues on pages 91 to 98).....	90
Table 6-2: SNMP Parameter (continues on pages 99 to 100).....	98
Table 6-3: SIP Configuration Parameters (continues on pages 101 to 111).....	100
Table 6-4: ISDN and CAS Interworking-Related Parameters (continues on pages 112 to 115).....	111
Table 6-5: Number Manipulation and Routing Parameters (continues on pages 116 to 122).....	115
Table 6-6: E1/T1/J1 Configuration Parameters (continues on pages 123 to 128).....	122
Table 6-7: Channel Parameters (continues on pages 129 to 132).....	128
Table 6-8: Configuration File Parameters.....	133
Table 8-1: Calling Name (Display).....	139
Table 8-2: Redirect Number.....	139
Table 8-3: Summary of DTMF Configuration Parameters (continues on pages 145 to 146).....	144
Table 8-4: Supported CDR Fields.....	151
Table 10-1: Vendor Specific Information Field.....	170
Table 10-2: Structure of the Vendor Specific Information Field.....	170
Table 12-1: Mediant 2000 Selected Technical Specifications (continues on pages 178 to 180).....	183
Table A-1: Mediant 2000 SIP Supplied Software Kit.....	187
Table B-1: Command Line Switch Descriptions.....	198
Table C-1: Packet Types Defined in RFC 1890.....	201
Table C-2: Defined Payload Types (continues on pages 196 to 197).....	201
Table C-3: Default RTP/RTCP/T.38 Port Allocation.....	202
Table F-1: Customizable Logo <i>ini</i> File Parameters.....	209
Table F-2: Web Appearance Customizable <i>ini</i> File Parameters.....	209
Table F-3: Customizable Logo <i>ini</i> File Parameters.....	210
Table F-4: Web Appearance Customizable <i>ini</i> File Parameters.....	210
Table I-1: Mapping of ISDN Release Reason to SIP Response (continues on pages 222 to 223).....	227
Table I-2: Mapping of SIP Response to ISDN Release Reason.....	229
Table J-1: SS7 Parameters (continues on pages 228 to 229).....	233
Table J-2: SIGTRAN Interface Groups (continues on pages 229 to 230).....	234
Table J-3: SIGTRAN Interface IDs.....	235
Table J-4: SS7 Signaling Link (continues on pages 231 to 232).....	236
Table J-5: Table of Parameter Values Example - Remote Management Connections.....	242
Table J-6: Table of Parameter Values Example - Port-to-Port Connections.....	242
Table K-1: General Mediant 2000 Parameters.....	249
Table K-2: VoiceXML Related Parameters.....	250
Table K-3: Supported RADIUS Attributes (continues on pages 246 to 247).....	251
Table K-4: VoiceXML Supported Elements & Attributes (continues on pages 251 to 255).....	256

---

Table K-5: VoiceXML Supported Properties .....	260
Table L-1: acBoardFatalError Alarm Trap .....	271
Table L-2: acBoardConfigurationError Alarm Trap.....	271
Table L-3: acBoardTemperatureAlarm Alarm Trap .....	272
Table L-4: acBoardEvResettingBoard Alarm Trap .....	272
Table L-5: acFeatureKeyError Alarm Trap .....	272
Table L-6: acBoardCallResourcesAlarm Alarm Trap .....	273
Table L-7: acBoardControllerFailureAlarm Alarm Trap .....	273
Table L-8: acBoardOverloadAlarm Alarm Trap .....	273
Table L-9: acActiveAlarmTableOverflow Alarm Trap .....	275
Table L-10: acBoardEthernetLinkAlarm Alarm Trap .....	275
Table L-11: coldStart Trap .....	276
Table L-12: authenticationFailure Trap.....	276
Table L-13: acBoardEvBoardStarted Trap .....	276



**Tip:** When viewing this manual on CD, Web site or on any other electronic copy, all cross-references are hyperlinked. Click on the page or section numbers (shown in blue) to reach the individual cross-referenced item directly. To return back to the point from where you accessed the cross-reference, press the **ALT** and ← keys.



**Note:** This User's Manual describes the Mediant 2000 SIP media gateway and the the TP-1610 SIP board.

## Trademarks

AC logo, Ardito, AudioCoded, AudioCodes, AudioCodes logo, IPmedia, Mediant, MediaPack, MP-MLQ, NetCoder, Stretto, TrunkPack, VoicePacketizer and VoIPerfect, are trademarks or registered trademarks of AudioCodes Limited. All other products or trademarks are property of their respective owners.

## Customer Support

Customer technical support and service are provided by AudioCodes' Distributors, Partners, and Resellers from whom the product was purchased. For Customer support for products purchased directly from AudioCodes, contact [support@audiocodes.com](mailto:support@audiocodes.com).

## Abbreviations and Terminology

Each abbreviation, unless widely used, is spelled out in full when first used. Only industry-standard terms are used throughout this manual. Hexadecimal notation is indicated by 0x preceding the number.

## Related Documentation

Document #	Manual Name
LTRT-690xx (e.g., LTRT-69001)	Mediant 2000 SIP Release Notes
LTRT-701xx	Mediant 2000 Fast Track Installation Guide



**Warning:** The Mediant 2000 is supplied as a sealed unit and must only be serviced by qualified service personnel.



**Note:** Where "network" appears in this manual, it means Local Area Network (LAN), Wide Area Network (WAN), etc. accessed via the gateway's Ethernet interface.

# 1 Overview

The Mediant 2000 SIP Voice over IP (VoIP) gateway enables voice, fax, and data traffic to be sent over the same IP network. The Mediant 2000 provides excellent voice quality and optimized packet voice streaming over IP networks.

The Mediant 2000 uses the award-winning, field-proven Digital Signal Processing (DSP) voice compression technology used in other TrunkPack™ series products.

The Mediant 2000 incorporates 1, 2, 4, 8 or 16 E1 or T1 spans for connection, directly to Public Switched Telephone Network (PSTN) / Private Branch Exchange (PBX) telephony trunks, and includes one or two 10/100 Base-TX Ethernet ports for connection to the network.

The Mediant 2000 supports up to 480 simultaneous VoIP or Fax over IP (FoIP) calls, supporting various Integrated Services Digital Network (ISDN) Primary Rate Interface (PRI) protocols such as EuroISDN, North American NI2, Lucent™ 4/5ESS, Nortel™ DMS100 and others. In addition, it supports different variants of Channel Associated Signaling (CAS) protocols for E1 and T1 spans, including MFC R2, E&M immediate start, E&M delay dial/start, loop start and ground start.

The Mediant 2000 gateway, best suited for large and medium-sized VoIP applications, is a compact device, comprising a 19-inch 1U chassis with optional dual AC or single DC power supplies.

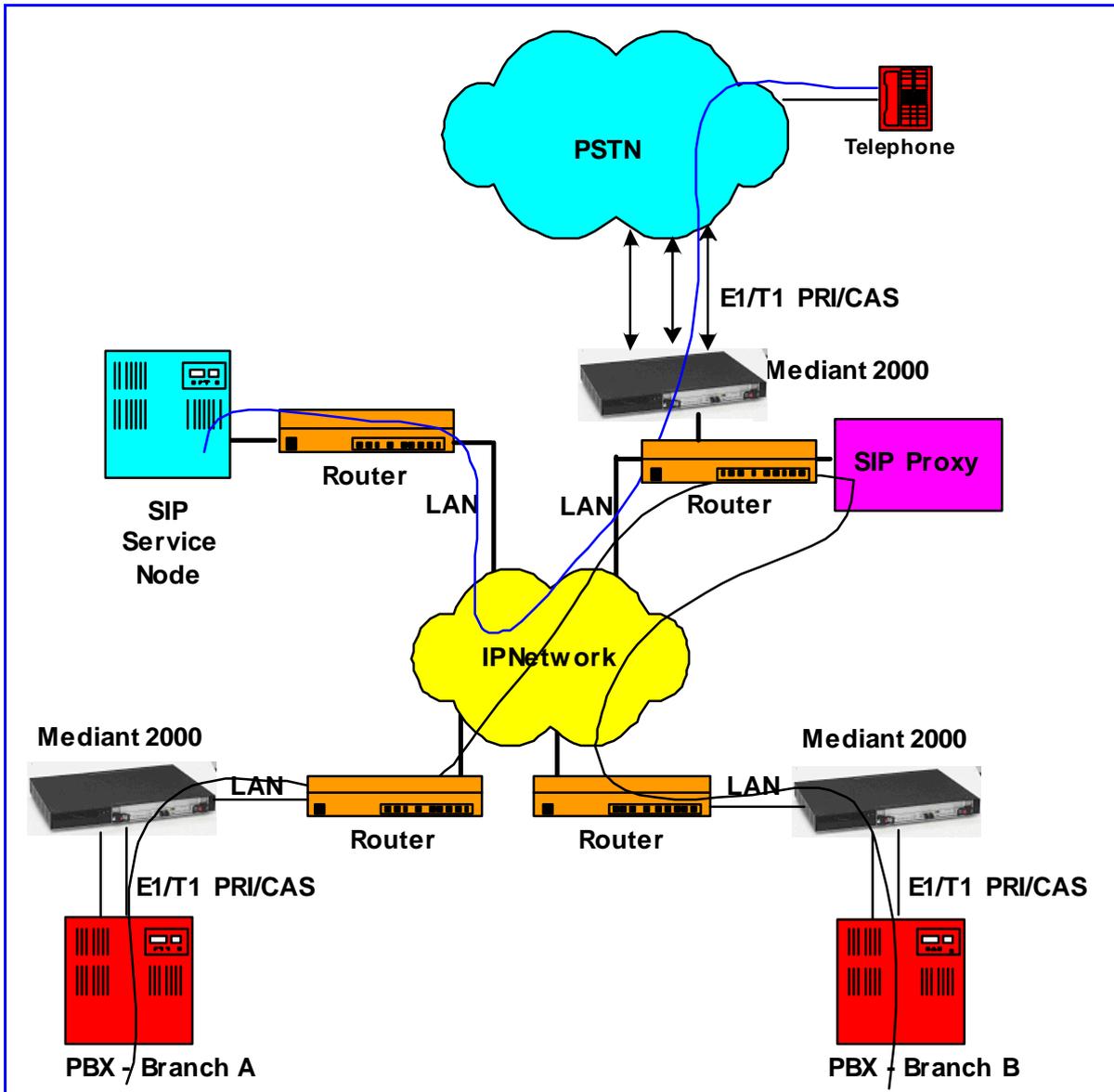
The deployment architecture can include several Mediant 2000 gateways in branch or departmental offices, connected to local PBXs. Call routing is performed by the gateways themselves or by SIP Proxy(s).

The Mediant 2000 gateway enables Users to make low cost long distance or international telephone/fax calls between distributed company offices, using their existing telephones/fax. These calls are routed over the existing network ensuring that voice traffic uses minimum bandwidth.

The Mediant 2000 can also route calls over the network using SIP signaling protocol, enabling the deployment of "Voice over Packet" solutions in environments where access is enabled to PSTN subscribers by using a trunking media gateway. This provides the ability to transmit voice and telephony signals between a packet network and a TDM network. Routing of the calls from the PSTN to a SIP service node (e.g., Call Center) is performed by the Mediant 2000 internal routing feature or by a SIP Proxy.

Figure 1-1 below illustrates typical Mediant 2000 gateway applications over VoIP Network.

Figure 1-1: Typical Mediant 2000 Gateway Application



## 1.1 Available Configurations

The Mediant 2000 is provided in the following configurations:

### E1 Available Configurations:

- 30 Channels on 1 E1 span with gateway-1 only
- 60 Channels on 2 E1 spans with gateway-1 only
- 120 Channels on 4 E1 spans with gateway-1 only
- 240 Channels on 8 E1 spans with gateway-1 only
- 480 Channels on 16 E1 spans with gateway-1 and gateway-2

### T1 Available Configurations:

- 24 Channels on 1 T1 span with gateway-1 only

- 48 Channels on 2 T1 spans with gateway-1 only
- 96 Channels on 4 T1 spans with gateway-1 only
- 192 Channels on 8 T1 spans with gateway-1 only
- 384 Channels on 16 T1 spans with gateway-1 and gateway-2

## 1.2 SIP Overview

SIP is an application-layer control (signaling) protocol used on the Mediant 2000 for creating, modifying, and terminating sessions with one or more participants. These sessions can include Internet telephone calls, media announcements and conferences.

SIP invitations are used to create sessions and carry session descriptions that enable participants to agree on a set of compatible media types. SIP uses elements called proxy servers to help route requests to the user's current location, authenticate and authorize users for services, implement provider call-routing policies and provide features to users.

SIP also provides a registration function that enables users to upload their current locations for use by proxy servers. SIP, on the Mediant 2000, complies with the IETF (Internet Engineering Task Force) RFC 3261 (refer to <http://www.ietf.org>).

## 1.3 Mediant 2000 Features

This section provides a high-level overview of some of the many Mediant 2000 supported features.

### 1.3.1 General Features

- Superior, high quality SIP PSTN gateway for Voice and fax over IP calls.
- Up to 16 E1/T1/J1 digital spans supporting various PRI and CAS protocols.
- Compliant with SIP (RFC 3261).
- Coders include: G.711, G.723.1, G.726, G.729A and NetCoder at 6.4 to 8.8 kbps, negotiable per channel.
- T.38 fax with superior performance (handling a round-trip delay of up to nine seconds).
- Echo Canceled with up to 128 msec tail length.
- Silence suppression with Comfort Noise Generation.
- Web management for easy configuration and installation.
- Simple Network Management Protocol (SNMP) and Syslog support.
- Simple Network Time Protocol (SNTP) support, the time-of-day can be obtained from a standard SNTP server.

### 1.3.2 Hardware Features

- Two 10/100 Base-TX Ethernet interface connections to the network, providing network redundancy.
- Compact, rugged 19-inch rack mount unit, one U high (1.75" or 44.5 mm), with two compactPCI™ (cPCI) slots.
- Optional cPCI slot for third-party CPU board.
- TP-1610/H.323 hot-swap cPCI board.
- Optional dual redundant AC or a single DC power supply.

### 1.3.3 PSTN-to-SIP Interworking

The Mediant 2000 gateway performs interworking between ISDN and CAS via E1/T1/J1 digital spans and SIP IETF signaling protocol. 16 E1, T1 or J1 spans are supported (480 channels) in a two modules gateway.

The Mediant 2000 gateway supports various ISDN PRI protocols such as EuroISDN, North American NI2, Lucent 4/5ESS, Nortel DMS100, Meridian 1 DMS100, Japan J1, as well as QSIG (basic call). PRI support includes User Termination or Network Termination side. ISDN-PRI protocols can be defined on an E1/T1 basis (i.e., different variants of PRI are allowed on different E1/T1 spans).

In addition, it supports numerous variants of CAS protocols for E1 and T1 spans, including MFC R2, E&M wink start, E&M immediate start, E&M delay dial/start, loop-start, and ground start. CAS protocols can be defined on an E1/T1 basis (i.e., different variants of CAS are allowed on different E1/T1 spans).

PSTN to SIP and SIP to PSTN Called number can be optionally modified according to rules that are defined in gateway *ini* file.

#### 1.3.3.1 Supported Interworking Features

- Definition and use of Trunk Groups for routing IP→PSTN calls.
- B-channel negotiation for PRI spans.
- ISDN Non Facility Associated Signaling (NFAS).
- PRI to SIP interworking according to draft-ietf-sipping-qsig2sip-04.txt.
- PRI to SIP Interworking of Q.931 Display (Calling name) information element.
- PRI (NI-2) to SIP interworking of Calling Name using Facility IE in Setup and Facility messages.
- Configuration of Numbering Plan and Type for IP→ISDN calls
- Interworking of PSTN to SIP release causes
- Interworking of ISDN redirect number to SIP diversion header (according to IETF draft-levy-sip-diversion-05.txt).
- Optional change of redirect number to called number for ISDN→ IP calls.
- Interworking of ISDN calling line Presentation & Screening indicators using RPID header <draft-ietf-sip-privacy-04.txt>.
- Interworking of Q.931 Called and Calling Number Type and Number Plan values using the RPID header.
- Supports ISDN en-block or overlap dialing for incoming Tel→IP calls.
- Supports routing of IP→Tel calls to predefined trunk groups.
- Supports a configurable channel select mode per trunk group.
- Supports various number manipulation rules for IP→Tel and Tel→IP, called and calling numbers.
- Option to configure ISDN Transfer Capability (per Gateway).

#### 1.3.4 Supported SIP Features

**The Mediant 2000 SIP main features are:**

- Reliable User Datagram Protocol (UDP) transport, with retransmissions.
- T.38 real time fax (using SIP).  
**Note:** If the remote side includes the fax maximum rate parameter in the SDP body of the Invite message, the gateway returns the same rate in the response SDP.

- Works with Proxy or without Proxy, using an internal routing table.
- Fallback to internal routing table if Proxy is not responding.
- Supports up to four Proxy servers. If the primary Proxy fails, the Mediant 2000 automatically switches to a redundant Proxy.
- Supports Proxy server discovery using Domain Name Server (DNS) SRV records.
- Proxy and Registrar Authentication (handling 401 and 407 responses) using Basic or Digest methods.
- Supported methods: INVITE, CANCEL, BYE, ACK, REGISTER, OPTIONS, INFO, REFER, UPDATE, NOTIFY, PRACK and SUBSCRIBE.
- Modifying connection parameters for an already established call (re-INVITE).
- Working with a Redirect server and handling 3xx responses.
- Early Media (supporting 183 Session Progress).
- PRACK reliable provisional responses <RFC 3262>.
- Call Hold and Transfer Supplementary services using REFER, Refer-To, Referred-By, Replaces and NOTIFY messages.
- Supports RFC 3327 – Adding “Path” to Supported header.
- Supports RFC 3581 – Symmetric Response Routing.
- Session Timer <draft-ietf-sip-session-timer-10.txt>.
- RFC 2833 Relay for Dual Tone Multi Frequency (DTMF) digits, including payload type negotiation.
- DTMF out-of-band transfer using:
  - INFO method <draft-choudhuri-sip-info-digit-00.txt>
  - INFO method, compatible with Cisco gateways
  - NOTIFY method <draft-mahy-sipping-signaled-digits-01.txt>. DTMF out-of-band transfer using INFO method (draft-choudhuri-sip-info-digit-00.txt)
- Can negotiate coder from a list of given coders.
- Supported coders:
  - G.711 A-law 64 kbps (10, 20, 30, 40, 50, 60, 80, 100, 120 msec)
  - G.711  $\mu$ -law 64 kbps (10, 20, 30, 40, 50, 60, 80, 100, 120 msec)
  - G.723.1 5.3, 6.3 kbps (30, 60, 90, 120 msec)
  - G.726 32 kbps (10, 20, 30, 40, 50, 60, 80, 100, 120 msec)
  - G.729A 8 kbps (10, 20, 30, 40, 50, 60, 80, 100 msec)  
G.729B is supported if Silence Suppression is enabled.
  - NetCoder 6.4, 7.2, 8.0 and 8.8 kbps (20, 40, 60, 80, 100, 120 msec).
  - EVRC\* 8, 4, 1 kbps (20, 40, 60, 80, 100, 120 msec)
  - AMR\* 4.75, 5.15, 5.90, 6.70, 7.40, 7.95, 10.2, 12.2 kbps (20 msec)
  - Transparent (20, 40, 60, 80, 100, 120 msec)

\* When EVRC (Enhanced Variable Rate Codec) and AMR (Adaptive Multi-Rate) are used, the number of available gateway channels is reduced (refer to the documentation of the parameter 'CoderName' in [Table 6-3](#)).

For more updated information on the gateway's supported features, refer to the latest Mediant 2000 & TP-1610 SIP Release Notes.

## Reader's Notes

## 2 Mediant 2000 Physical Description

This section provides detailed information on the Mediant 2000 hardware components, the location and functionality of the LEDs, buttons and connectors on the front and rear panels.

### 2.1 General

The Mediant 2000 gateway comprises the following hardware components:

- A 19-inch 1U high rack mount chassis (refer to Section 2.2 on page 20).
- A single compactPCI™ TP-1610 board (refer to Section 2.3 on page 20).
- A single TP-1610 Rear Transition Module (RTM) (refer to Section 2.4 on page 24).
- A single available cPCI slot for an optional third-party CPU board (refer to Section 2.5 on page 25).

Figure 2-1 shows the front view of the Mediant 2000 media gateway.

Figure 2-1: Mediant 2000 Front View

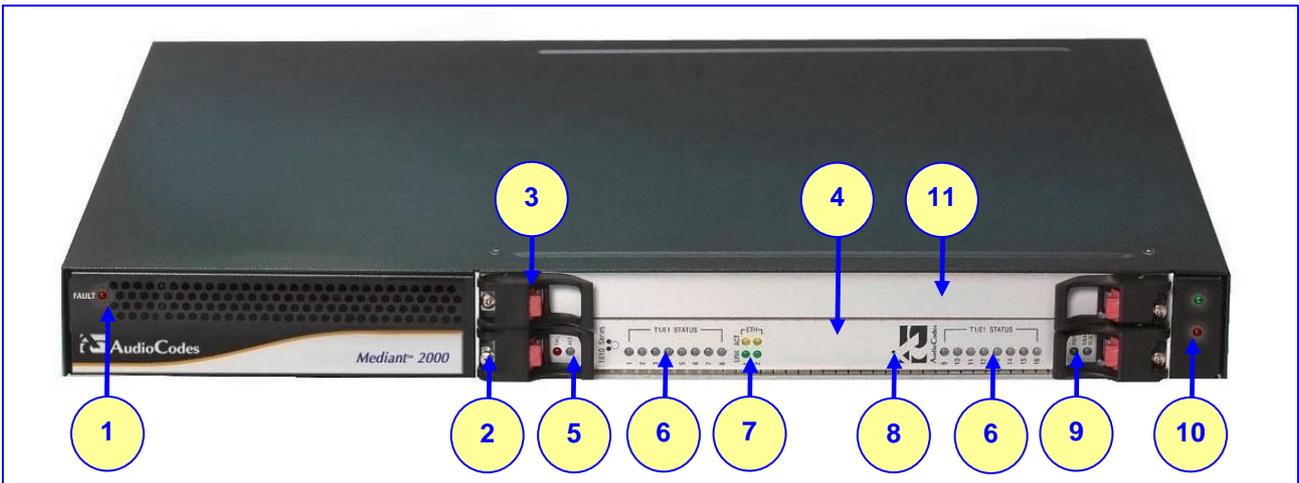


Table 2-1: Mediant 2000 Front View Component Descriptions

Item #	Label	Component Description
1	FAULT	Dual AC Power LED.
2		cPCI board locking screws.
3		cPCI latches.
4		TP-1610 cPCI board, 16-trunk configuration.
5		Status LED Indicators.
6	T1/E1 STATUS	E1/T1 Trunk Status LED Indicators.
7	ETH	Ethernet LED Indicators.
8		Reset button.
9		cPCI LED Indicators.
10		Power and Fan LEDs
11		An available cPCI slot for an optional third-party CPU board.

## 2.2 The Mediant 2000 Chassis

The Mediant 2000 chassis is an industrial platform, 19" wide, 1U high and 12" deep that houses the TP-1610 board in its front cage, slot #1 (the lower slot) and the TP-1610 RTM in its rear cage, slot #1 (the lower slot).

Slot # 2 in the Mediant 2000 chassis' front and rear cages can optionally be used by customers for a CPU board.

Refer to [Table 2-2](#) for detailed description of the chassis' LED indicators.

**Table 2-2: Chassis LED Indicators**

Location	Color	Function
Right side of front panel	Green	The power is on.
Right side of front panel	Red	Fan failure - indicates that any of the internal fans has significantly reduced its speed or has frozen.
Left side of front panel	Red	Power supply failure - indicates that one of the two AC redundant power supplies is faulty or disconnected from the AC/mains outlet. (This LED is only relevant for the dual AC power supply).

### 2.2.1 Power Supply

The Mediant 2000 power supply is available in three configuration options:

- Single universal 100-240 VAC 1 A max, 50-60 Hz.
- Dual-redundant 100-240 VAC 1.5 A max, 50-60 Hz.
- -48 VDC power supply suitable for field wiring applications.

## 2.3 The TP-1610 Board

The Mediant 2000 is populated by a single compactPCI™ board, the TP-1610 (shown in [Figure 2-2](#)). The TP-1610 is a high-density, hot-swappable, cPCI resource board with a capacity of up to 480 ports, supporting all necessary functions for voice, data and fax streaming over IP networks. The TP-1610 is composed of one or two identical media gateways modules: Gateway-1 and Gateway-2, each containing 240 DSP channels. These media gateways are fully independent, each gateway having its own MAC (Media Access Control) and IP addresses and LED indicators. The TP-1610 board is supplied with a rear I/O configuration in which both PSTN trunks and Ethernet interface are located on a passive rear I/O module (for information on the RTM, refer to [Section 2.4](#) on page 24).

Figure 2-2: Front and Upper View of the TP-1610 cPCI Board

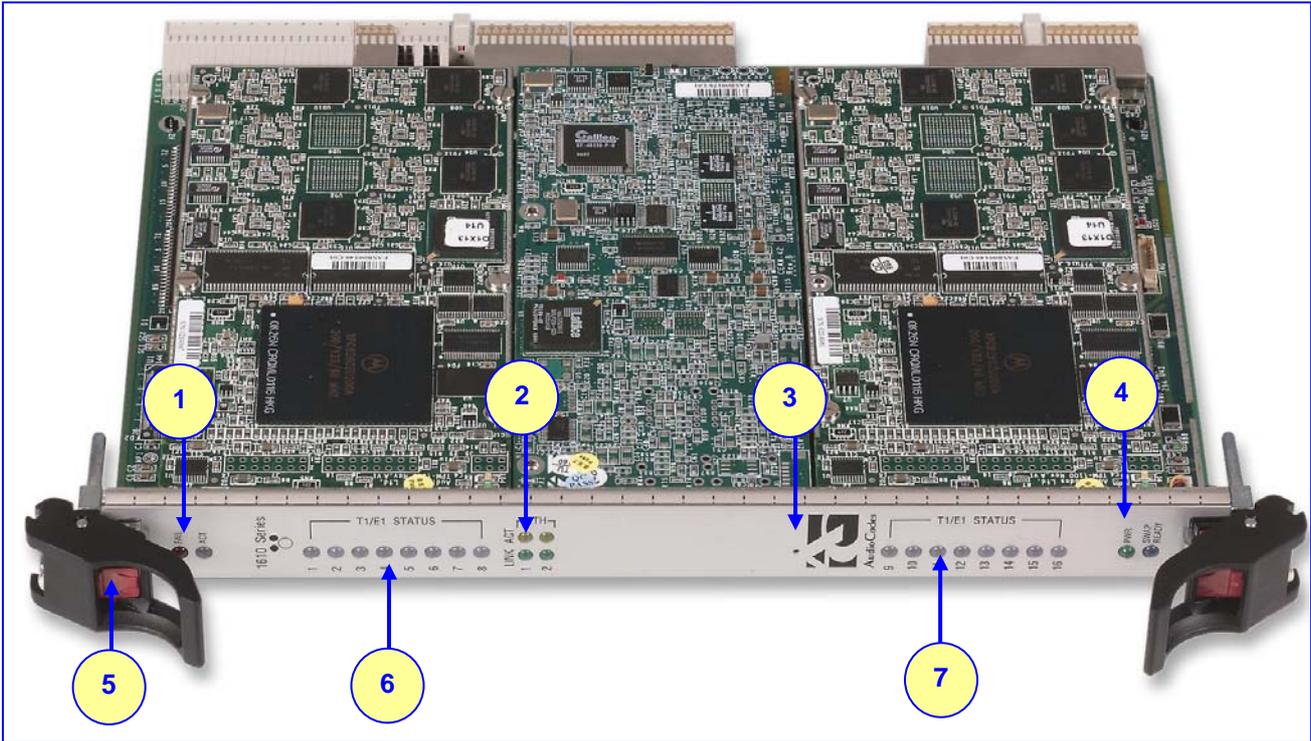


Table 2-3: Front and Upper View of the TP-1610 cPCI Board Component Descriptions

Item #	Label	Component Description
1		Status LEDs
2	ETH	Ethernet LEDs
3		Reset button
4		cPCI LEDs
5		cPCI Latch
6	T1 / E1 STATUS	T1/E1 Trunk Status LEDs (for each of trunks 1 to 8)
7	T1 / E1 STATUS	T1/E1 Trunk Status LEDs (for each of trunks 9 to 16)

### 2.3.1 Board Hot-Swap Support

The TP-1610 cPCI board is hot-swappable and can therefore be removed from a slot (and inserted into a slot) while the Mediant 2000 is under power. It is recommended though that you power down the chassis and read the notes below before replacing the components.

For details on removing/inserting the optional CPU board, refer to the directions accompanying it.



#### Electrical Component Sensitivity

Electronic components on printed circuit boards are extremely sensitive to static electricity. Normal amounts of static electricity generated by clothing can damage electronic equipment. To reduce the risk of damage due to electrostatic discharge when installing or servicing electronic equipment, it is recommended that anti-static earthing straps and mats be used.



**Note 1:** Before removing or inserting boards from / to the chassis, attach a wrist strap for electrostatic discharge (ESD) and connect it to the rack frame using an alligator clip.

**Note 2:** Do not set components down without protecting them with a static bag.

### 2.3.1.1 Removing Boards

➤ **To remove the TP-1610 board from the chassis, take these 3 steps:**

1. Unfasten the screws on the plate of the board.
2. Press the red ejector buttons on the two black ejector/injector latches on both ends and wait for the hot-swap blue LED to light, indicating that the board can be removed.
3. Pull on the two ejector/injector latches and ease out the board from the slot.

➤ **To remove the TP-1610 RTM from the chassis, take these 4 steps:**

1. Remove the cables attached to the RTM.
2. Unfasten the screws on the brackets at both ends of the panel that secure the RTM to the chassis.
3. Press the red ejector buttons on the two black ejector/injector latches on both ends.
4. Grasp the panel and ease the RTM board out of the slot.

### 2.3.1.2 Inserting Boards

➤ **To insert the TP-1610 board into the chassis, take these 6 steps:**

1. Hold the board horizontally.
2. With the black ejector/injector latches in the open (pulled out) position, insert the board in the slot, aligning the board with the grooves on each end.
3. Ease the board all the way into the slot until the ejector/injector latches touch the chassis. The Blue hot-swap LED is lit.
4. Press the two black ejector/injector latches on both ends inward, toward the middle, until you hear a click.
5. Wait for the hot-swap blue LED to turn off.
6. Fasten the screws on the front panel of the board to secure the board to the chassis and to ensure that the board has a chassis earthing connection.

➤ **To insert the TP-1610 RTM into the chassis, take these 6 steps:**

1. Hold the board horizontally.
2. With the black ejector/injector latches in the open (pulled out) position, insert the board in the slot, aligning the board with the grooves on each end.
3. Ease the board all the way into the slot until the ejector/injector latches touch the chassis.
4. Press the two black ejector/injector latches on both ends inward, toward the middle until you hear a click.
5. Fasten the screws on the front panel of the board to secure the board to the chassis and to ensure that the board has a chassis earthing connection.
6. Reattach the cables (refer to Section 3.4 on page 30).

### 2.3.2 TP-1610 Front Panel LED Indicators

The functionality of the front panel LEDs for the TP-1610 is described in the following four tables and illustrated in [Figure 2-2](#) on page 21. Note that there is a choice of front panels according to the number of channels.

**Table 2-4: Status LED Indicators**

Label	LED Color	LED Function	
FAIL	Red	Normally OFF; Red indicates gateway failure (fatal error)	
ACT	Green	Gateway initialization sequence terminated OK	
	Yellow	N/A	

**Table 2-5: E1/T1 Trunk Status LED Indicators**

Label	LED Color	Signal Description	
T1/E1 Status 1 to 8 and T1/E1 Status 9 to 16	Green	Trunk is synchronized (normal operation)	
	Red	Loss due to any of the following 4 signals:	
		LOS	Loss of Signal
		LFA	Loss of Frame Alignment
		AIS	Alarm Indication Signal (the Blue Alarm)
		RAI	Remote Alarm Indication (the Yellow Alarm)



**Note:** On the front panel 16 LEDs are provided for 16-span units and 8 LEDs are provided for 1-span, 2-span, 4-span, and 8-span units. In the case of 1-span, 2-span and 4-span units, the extra LEDs are unused.

**Table 2-6: Ethernet LED Indicators**

Label	LED Color	LED Function
LINK	Green	Link all OK
ACT	Yellow	Transmit / receive activity

**Table 2-7: cPCI LED Indicators**

Label	LED Color	LED Function
PWR	Green	Power is supplied to the board
SWAP READY	Blue	The cPCI board can now be removed.
		The cPCI board was inserted successfully. For detailed information on the Swap-Ready LED, refer to <a href="#">Section 2.3.1</a> on page 21.

During correct Mediant 2000 operation, the ACT LED is lit green, the FAIL LED is off. Changing of the FAIL LED to red indicates a failure.

## 2.4 Rear Transition Module

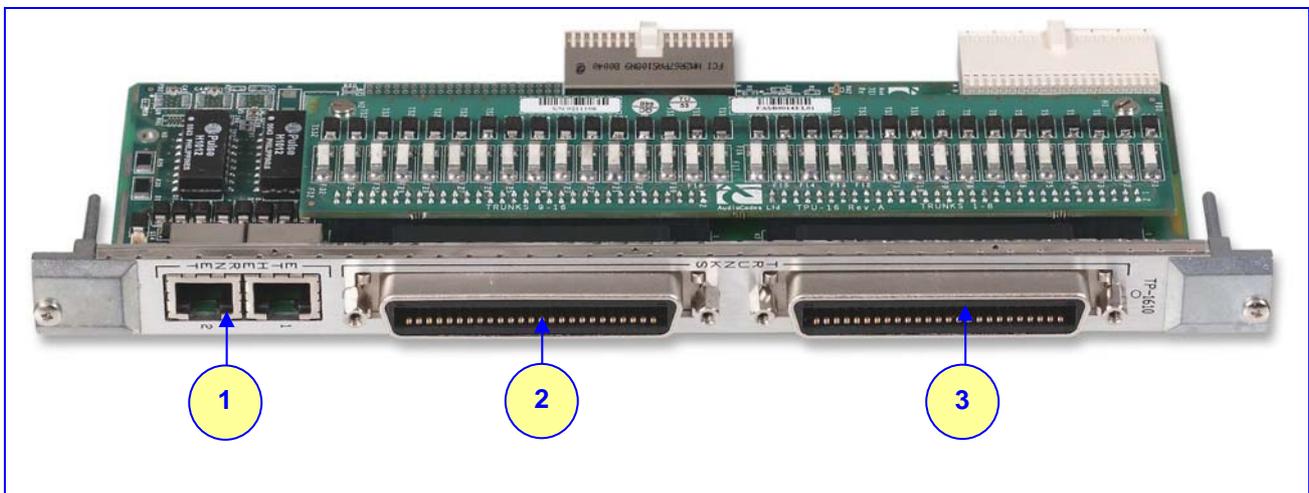
The Mediant 2000 RTM includes a PSTN trunks and an Ethernet interfaces.

The Ethernet interface features dual 10/100 Base-TX, RJ-45 shielded connectors for (an active / standby) redundancy scheme providing protection against the event of a failure.

The PSTN interface is provided with a choice of rear panels (1-span, 2-span, 4-span, 8-span or 16-span).

Rear panel with two 50-pin female Telco connectors (DDK 57AE-40500-21D) (shown in [Figure 2-3](#)) is required for a gateway equipped with up to 16 E1/T1 spans. Rear panel with RJ-48c connectors (shown in [Figure 2-4](#)) is required for a gateway equipped with 1, 2, 4, or 8 E1/T1 spans. The physical difference between the 1-Span, 2-Span and 4-Span RTMs, and the 8-span RTM is that the RJ-48c ports are depopulated correspondingly.

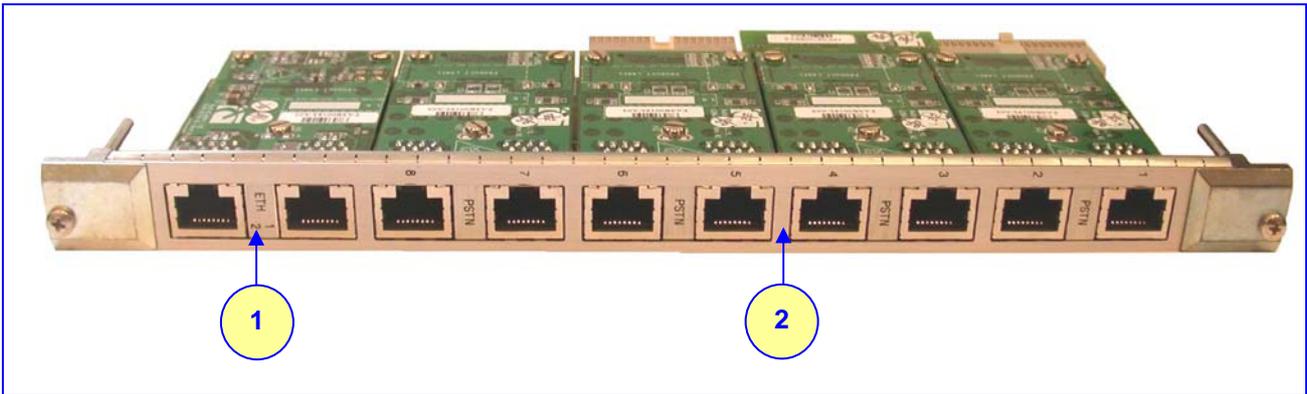
**Figure 2-3: Rear Panel with two 50-pin Connectors for 16 Trunks**



**Table 2-8: Rear Panel with two 50-pin Connectors for 16 Trunks Component Descriptions**

Item #	Label	Component Description
1	ETHERNET	2 Ethernet Ports. 2 RJ-45 network connectors.
2	TRUNKS	E1/T1 trunks 9 to 16. 50-pin male Telco connector.
3	TRUNKS	E1/T1 trunks 1 to 8. 50-pin male Telco connector.

**Figure 2-4: Rear Panel with 8 RJ-48c Connectors for 8 Trunks**



**Table 2-9: Rear Panel with 8 RJ-48c Connectors for 8 Trunks Component Descriptions**

Item #	Label	Component Description
1	ETHERNET	2 Ethernet Ports. 2 RJ-45 network connectors
2	TRUNKS	8 E1/T-1 Spans. 8 RJ-48c trunk connectors

## 2.5 Optional CPU Board

The Mediant 2000 provides an optional second cPCI slot that can be optionally used for customer's CPU board. This CPU board can be used for general applications such as a Gatekeeper, Softswitch, Application Server or other. The following CPU boards were tested for compliancy with the Mediant 2000 chassis:

- Sun™: CP2080 + PMC-233 (Ramix™ disk on board) + Rear Transition Module (RTM).
- Intel™ ZT5515B-1A with 40GB on-board disk plus RTM (ZT4807).

## Reader's Notes

## 3 Installing the Mediant 2000

This section describes the hardware installation procedures for the Mediant 2000. For information on how to start using the gateway, refer to Section 4 on page 35. For detailed information on the Mediant 2000 connectors, LEDs and buttons, refer to Section 2 on page 19.



### Caution Electrical Shock

The equipment must only be installed or serviced by qualified service personnel.

➤ **To install the Mediant 2000, take these 4 steps:**

1. Unpack the Mediant 2000 (refer to Section 3.1 below).
2. Check the package contents (refer to Section 3.2 below).
3. Mount the Mediant 2000 (refer to Section 3.3 on page 28).
4. Cable the Mediant 2000 (refer to Section 3.4 on page 30).

After powering-up the Mediant 2000, the Ready and LAN LEDs on the front panel turn to green (after a self-testing period of about 3 minutes). Any malfunction changes the Ready LED to red (refer to Section 2.3.2 on page 23 for details on the Mediant 2000 LEDs).

When you have completed the above relevant sections you are then ready to start configuring the gateway (Section 4 on page 35).

### 3.1 Unpacking

➤ **To unpack the Mediant 2000, take these 6 steps:**

1. Open the carton and remove packing materials.
2. Remove the Mediant 2000 gateway from the carton.
3. Check that there is no equipment damage.
4. Check, retain and process any documents.
5. Notify AudioCodes or your local supplier of any damage or discrepancies.
6. Retain any diskettes or CDs.

### 3.2 Package Contents

Ensure that in addition to the Mediant 2000, the package contains:

- For the dual AC power supply version two AC power cables are supplied; for the single AC power supply version one AC power cable is supplied.
- For the DC power supply version, one connectorized DC power cable (crimp connection type) and one DC adaptor (screw connection type) connected to the rear panel of the Mediant 2000 are supplied; use only one type.
- CD (software and documentation).
- Small plastic bag containing (refer to Figure 3-1):
  - Two brackets and four bracket-to-device screws for 19-inch rack installation option.
  - Four anti-slide bumpers for desktop / shelf installation option.
- The Mediant 2000 Fast Track Installation Guide.

**Figure 3-1: 19-inch Rack & Desktop Accessories**

## 3.3 Mounting the Mediant 2000

The Mediant 2000 can be mounted on a desktop, or installed in a standard 19-inch rack. Refer to Section 3.4 on page 30 for cabling the Mediant 2000.

### 3.3.1 Mounting the Mediant 2000 on a Desktop

No brackets are required. Optionally, attach the four (supplied) anti-slide bumpers to the base of the Mediant 2000 and place it on the desktop in the position you require.

### 3.3.2 Installing the Mediant 2000 in a 19-inch Rack

Users can install the device in a standard 19-inch rack either by placing the device on a shelf preinstalled in the rack (preferred method), or by attaching the device directly to the rack's frame via integral brackets.

Before rack mounting the chassis, attach the two (supplied) brackets to the front sides of the device (refer to Figure 3-2).

➤ **To attach the two front side brackets, take these 3 steps:**

1. Remove the 2 screws nearest the front panel on either side of the device.
2. Align a bracket over 2 holes on one side (so that the bracket's larger holes face front) and with the 2 supplied replacement screws, screw in the bracket.
3. Perform the same procedure on the other side.

Figure 3-2: Mediant 2000 Front View with 19-inch Rack Mount Brackets



### Rack Mount Safety Instructions (UL)

When installing the chassis in a rack, be sure to implement the following Safety instructions recommended by Underwriters Laboratories:



- **Elevated Operating Ambient** - If installed in a closed or multi-unit rack assembly, the operating ambient temperature of the rack environment may be greater than room ambient. Therefore, consideration should be given to installing the equipment in an environment compatible with the maximum ambient temperature (T<sub>ma</sub>) specified by the manufacturer.
- **Reduced Air Flow** - Installation of the equipment in a rack should be such that the amount of air flow required for safe operation on the equipment is not compromised.
- **Mechanical Loading** - Mounting of the equipment in the rack should be such that a hazardous condition is not achieved due to uneven mechanical loading.
- **Circuit Overloading** - Consideration should be given to the connection of the equipment to the supply circuit and the effect that overloading of the circuits might have on overcurrent protection and supply wiring. Appropriate consideration of equipment nameplate ratings should be used when addressing this concern.
- **Reliable Earthing** - Reliable earthing of rack-mounted equipment should be maintained. Particular attention should be given to supply connections other than direct connections to the branch circuit (e.g., use of power strips.)

#### ➤ To attach the device to a 19-inch rack, take these 2 steps:

1. Position the device in your 19-inch rack and align the left-hand and right-hand bracket holes to holes (of your choosing) in the vertical tracks of the 19-inch rack.
2. Use standard 19-inch rack bolts (not provided) to fasten the device to the frame of the rack.

AudioCodes recommends using two additional (not supplied) rear mounting brackets to provide added support.



**Note:** Users assembling the rear brackets by themselves should note the following:

- The distance between the screws on each bracket is 26.5 mm.
- To attach the brackets, use 4-40 screws with a maximal box penetration length of 3.5 mm.

#### ➤ To place the device on a 19-inch rack's shelf, take these 2 steps:

1. Place the device on the preinstalled shelf.

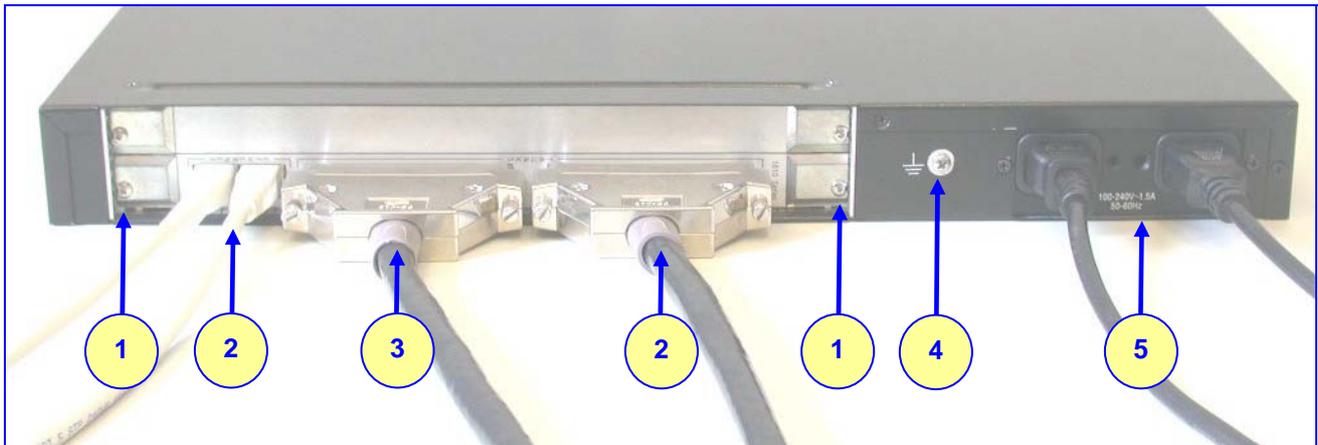
2. You're now recommended to take the optional steps of fastening the device to the frame of the rack (as described above) while it is placed on the shelf, so preventing it from sliding when inserting cables into connectors on the rear panel.

### 3.4 Cabling the Mediant 2000

Refer to Section 2 on page 19 for detailed information on the Mediant 2000 rear panel connectors and LEDs.

Note that the Mediant 2000 is available in many *configurations*, i.e., AC or DC, in the 16-trunk, 8-trunk, 4-trunk, 2-trunk or 1-trunk device. The 16-trunk dual AC (Figure 3-3) and the 8-trunk DC (Figure 3-4) configurations are illustrated here as *representative* products.

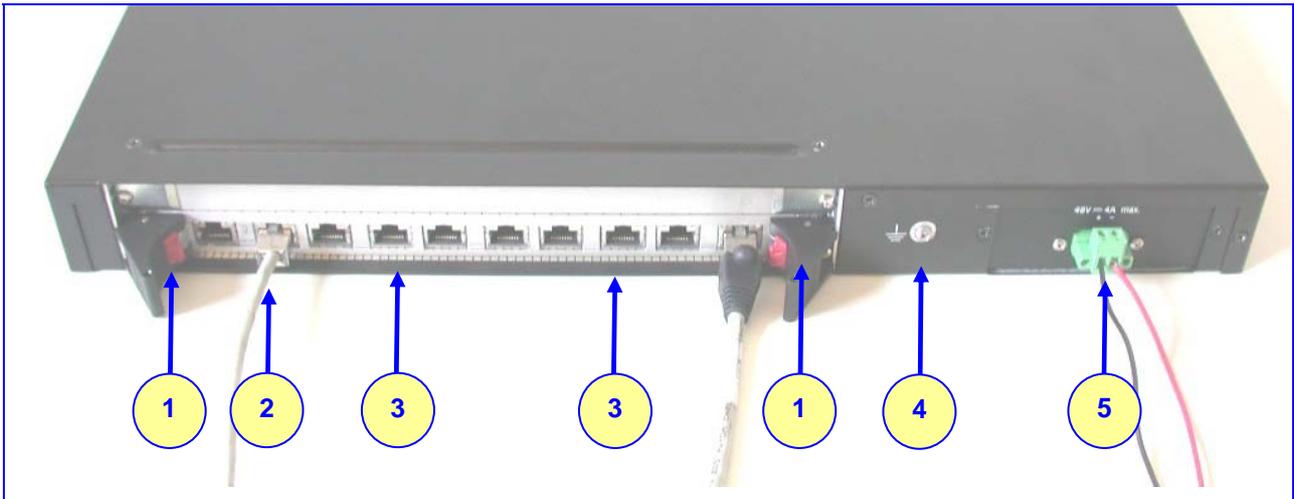
**Figure 3-3: Mediant 2000 Rear Panel Cabling (16 Trunks, Dual AC Power)**



**Table 3-1: Mediant 2000 Rear Panel Cabling (16 Trunks, Dual AC Power) Component Descriptions**

Item #	Label	Component Description
1		RTM locking screws.
2	ETHERNET	Two Category 5 network cables, connected to the 2 Ethernet RJ-45 ports.
3	TRUNKS	Two 50-pin Telco connector cables, each supporting 8 trunks.
4		Protective earthing screw.
5	100-240~1.5A	Dual AC power cables.

**Figure 3-4: Mediant 2000 Rear Panel Cabling (8 Trunks, DC Power))**



**Table 3-2: Mediant 2000 Rear Panel Cabling (8 Trunks, DC Power) Component Descriptions**

Item #	Label	Component Description
1		RTM latches.
2	ETH	A Category 5 network cable, connected to the Ethernet 1 RJ-45 port.
3	PSTN	8 RJ-48c ports, each supporting a trunk.
4		Protective earthing screw.
5	48V 4A max	2-pin connector for DC.



### Electrical Earthing

The unit must be permanently connected to earth via the screw provided at the back on the unit. Use 14-16 AWG wire and a proper ring terminal for the earthing.

➤ **To cable the Mediant 2000, take these 4 steps:**

1. Permanently connect the device to a suitable earth with the protective earthing screw on the rear connector panel, using 14-16 AWG wire.
2. Connect the E1/T1 trunk interfaces (refer to Section 3.4.1 below).
3. Install the Ethernet connection (refer to Section 3.4.2 on page 32).
4. Connect the power supply (refer to Section 3.4.3 on page 33).

### 3.4.1 Connecting the E1/T1 Trunk Interfaces

Connect the Mediant 2000 E1/T1 Trunk Interfaces using **either** Telco or RJ-48 connectors:

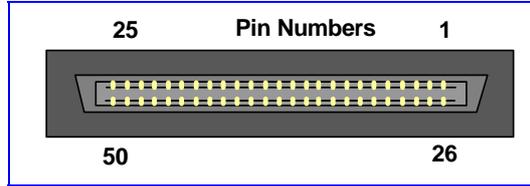
➤ **With 50-pin Telco connectors (16-trunk device), take these 3 steps:**

1. Attach the Trunk cable with a 50-pin male Telco connector to the 50-pin female Telco connector labeled "Trunks 1→8" on the Rear Transition Module (RTM).
2. Connect the other end of the Trunk cable to the PBX/PSTN switch.

- Repeat steps 1 and 2 for the other Trunk cable but this time connect it to the connector labeled “Trunks 9→16”.

The 50-pin male Telco cable connector must be wired according to the pinout in [Table 3-3](#) below, and to mate with the female connector illustrated in [Figure 3-5](#).

**Figure 3-5: 50-pin Female Telco Board-Mounted Connector**



**Table 3-3: E1/T1 Connections on each 50-pin Telco Connector**

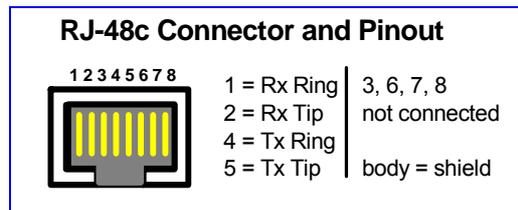
E1/T1 Number		Tx Pins (Tip/Ring)	Rx Pins (Tip/Ring)
1 to 8	9 to 16		
1	9	27/2	26/1
2	10	29/4	28/3
3	11	31/6	30/5
4	12	33/8	32/7
5	13	35/10	34/9
6	14	37/12	36/11
7	15	39/14	38/13
8	16	41/16	40/15

➤ **With RJ-48c Connectors, take these 2 steps:**

- Connect the E1/T1 trunk cables to the ports labeled “Trunks 1 to 8” (in the case of the 8-trunk device) on the Mediant 2000 RTM.
- Connect the other ends of the Trunk cables to the PBX/PSTN switch.

RJ-48c trunk connectors are wired according to [Figure 3-6](#) below.

**Figure 3-6: Pinout of RJ-48c Trunk Connectors**



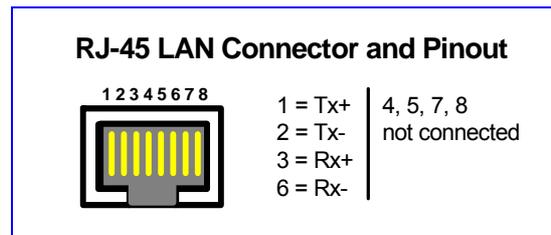
### 3.4.2 Installing the Ethernet Connection

Connect a standard Category 5 network cable to the Ethernet RJ-45 port (and the other as optional redundancy/backup). Connect the other end of the Category 5 network cables to your IP network. The Ethernet connectors (labeled Ethernet 1 and Ethernet 2) are wired according to [Figure 3-7](#).

Note that for redundant operation it is recommended to connect each of the Ethernet connectors to a different Switch.

When assigning an IP address to the Mediant 2000 using HTTP (under Step 1 in Section 4.1.1), you may be required to disconnect this cable and re-cable it differently.

Figure 3-7: Pinout of RJ-45 Connectors



### 3.4.3 Connecting the Power Supply

Connect the Mediant 2000 to the power supply using one of the following methods:

#### 3.4.3.1 Connecting the AC Power Supply

➤ **When using a single AC power cable:**

Attach one end of the supplied 100/240 VAC power cable to the rear AC socket and connect the other end to the correct earthed AC power supply.

➤ **When using a dual AC power cable:**

Attach one end of the supplied 100/240 VAC power cables to the rear AC sockets and connect the other end to a separate earthed mains circuits (for power source redundancy).



**Note:** For the dual AC power supply note the following:

- The LED on the left side of the chassis is only connected when the dual AC is used. It is not relevant to the single AC power connection.
- If only a single socket is connected to the AC power, (while the other plug is left unconnected) the chassis' LED (on the left side) is lit Red, indicating that one of the dual power inlets is disconnected.
- When both the AC power cables are connected, one of the plugs can be disconnected under power without affecting operation, in which case the chassis' left LED is lit Red.
- UPS can be connected to either (or both) of the AC connections.
- The dual AC connections operate in a 1 + 1 configuration and provide load-sharing redundancy.
- Each of the dual power cables can be connected to different AC power phases.

#### 3.4.3.2 Connecting the DC Power Supply

To connect the Mediant 2000 to a DC power supply use one of these two options:

- DC Terminal block with a screw connection type.
- DC Terminal block with a crimp connection type.

➤ **When using a DC terminal block screw connector, take these 3 steps:**

1. Create a DC cable by inserting two 14-16 AWG insulated wires into the supplied adaptor (refer to [Figure 3-8](#)) and fasten the two screws, each one located directly above each wire.
2. Connect the two insulated wires to the correct DC power supply. Ensure that the connections to the DC power supply maintain the correct polarity.
3. Insert the terminal block into the DC inlet located on the Mediant 2000.

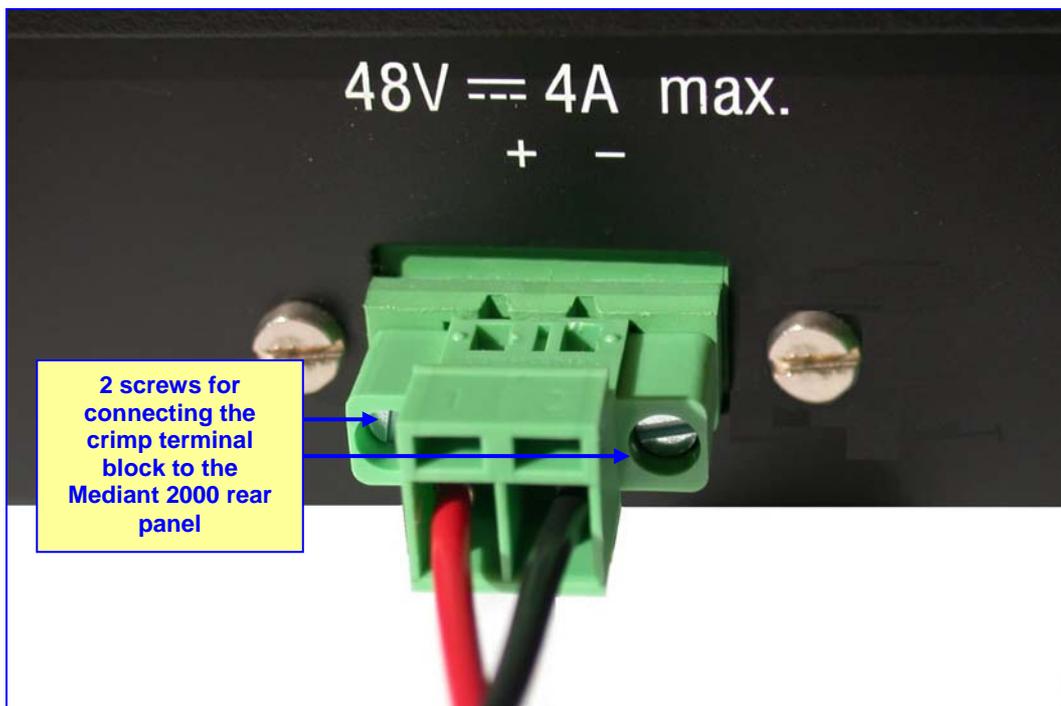
Figure 3-8: DC Terminal Block Screw Connector



➤ **When using a DC terminal block crimp connector, take these 3 steps:**

1. Remove the DC adaptor (screw connection type) that is attached to the Mediant 2000 rear panel.
2. Connect the two insulated wires to the correct DC power supply. Ensure that the connections to the DC power supply maintain the correct polarity (refer to [Figure 3-9](#)).
3. Insert the terminal block into the DC inlet located on the Mediant 2000.

Figure 3-9: DC Terminal Block Crimp Connector



## 4 Getting Started

The Mediant 2000 is supplied with application software already resident in its flash memory (with factory default parameters).

Section 4.1 below describes how to assign IP addresses to the Mediant 2000, while Section 4.1.2 on page 35 describes how to set up the Mediant 2000 with basic parameters using a standard Web browser (such as Microsoft™ Internet Explorer).

For detailed information on how to *fully* configure the gateway, refer to the Web Interface, described in Section 5 on page 39.

### 4.1 Assigning the Mediant 2000 IP Address

The Mediant 2000 is composed of one or two identical media gateway modules. These media gateways are fully independent, each gateway having its own MAC and IP addresses (Table 4-1 shows the default IP addresses of the Mediant 2000). To assign an IP address to each of the Mediant 2000 modules use one of the following methods:

- HTTP using a Web browser (refer to Section 4.1.1 below).
- BootP (refer to Section 4.1.2 on page 36).
- Dynamic Host Control Protocol (DHCP) (refer to Section 10.2 on page 169).

You can use the 'Reset' button to restore the Mediant 2000 networking parameters to their factory default values (refer to Section 4.2 on page 36).

**Table 4-1: Mediant 2000 Default Networking Parameters**

Mediant 2000 Version	Default Value
Single module (up to 8 Trunks)	10.1.10.10
Double module (up to 16 Trunks)	10.1.10.10 (Trunks 1-8) and 10.1.10.11 (Trunks 9-16)
Default subnet mask is 255.255.0.0, default gateway IP address is 0.0.0.0	

#### 4.1.1 Assigning an IP Address Using HTTP

➤ **To assign an IP address using HTTP, take these 9 steps:**

1. Disconnect the Mediant 2000 from the network and reconnect it to your PC using one of the following two methods:
  - Use a standard Ethernet cable to connect the network interface on your PC to a port on a network hub / switch. Use a second standard Ethernet cable to connect the Mediant 2000 to another port on the same network hub / switch.
  - Use an Ethernet cross-over cable to directly connect the network interface on your PC to the Mediant 2000.
2. Change your PC's IP address and subnet mask to correspond with the Mediant 2000 factory default IP address and subnet mask, shown in Table 4-1. For details on changing the IP address and subnet mask of your PC, refer to Windows™ Online Help (Start>Help).
3. Access the Mediant 2000 first module's Embedded Web Server (refer to Section 5.6 on page 41).
4. In the 'Quick Setup' screen (shown in Figure 4-1), set the Mediant 2000 'IP Address', 'Subnet Mask' and 'Default Gateway IP Address' fields under 'IP Configuration' to

correspond with your network IP settings. If your network doesn't feature a default gateway, enter a dummy value in the 'Default Gateway IP Address' field.

5. Click the Reset button and click OK in the prompt; The Mediant 2000 applies the changes and restarts. This takes approximately 3 minutes to complete. When the Mediant 2000 has finished restarting, the Ready and LAN LEDs on the front panel are lit green.



**Tip:** Record and retain the IP address and subnet mask you assign the Mediant 2000. Do the same when defining new username or password. If the Embedded Web Server is unavailable (for example, if you've lost your username and password), use the BootP/TFTP (Trivial File Transfer Protocol) configuration utility to access the device, "reflash" the load and reset the password (refer to [Appendix B](#) on page 189 for detailed information on using a BootP/TFTP configuration utility to access the device).

6. Repeat steps 3 to 5 for the Mediant 2000 second module (if used).
7. Disconnect your PC from the Mediant 2000 or from the hub / switch (depending on the connection method you used in step [Error! Reference source not found.](#)).
8. Reconnect the Mediant 2000 and your PC (if necessary) to the network.
9. Restore your PC's IP address and subnet mask to what they originally were. If necessary, restart your PC and re-access the Mediant 2000 via the Embedded Web Server with its new assigned IP address.

### 4.1.2 Assigning an IP Address Using BootP



**Note:** BootP procedure can also be performed using any standard compatible BootP server.



**Tip:** You can also use BootP to load the auxiliary files to the Mediant 2000 (refer to Section [6.12.1](#) on page 132).

➤ **To assign an IP address using BootP, take these 4 steps:**

1. Open the BootP application (supplied with the Mediant 2000 software package).
2. Add client configuration for the gateway that you want to initialize, refer to Section [B.11.1](#) on page 195.
3. Reset the gateway physically causing it to use BootP; the Mediant 2000 changes its network parameters to the values provided by the BootP.
4. Repeat steps 2 and 3 for the Mediant 2000 second module (if used).

## 4.2 Restoring Networking Parameters to their Initial State

You can use the 'Reset' button to restore the Mediant 2000 networking parameters to their factory default values (described in [Table 4-1](#)) and to reset the username and password.

Note that the Mediant 2000 returns to the software version burned in flash. This process also restores the Mediant 2000 parameters to their factory settings, therefore you must load your previously backed-up *ini* file, or the default *ini* file (received with the software kit) to set them to their correct values.

This option is currently supported on one media gateway module (trunks 1-8) only.

➤ **To restore networking parameters to their initial state, take these 6 steps:**

1. Disconnect the Mediant 2000 from the power and network cables.
2. Reconnect the power cable; the gateway is powered up. After approximately 45 seconds the ACT LED blinks for about 4 seconds.
3. While the ACT LED is blinking, press shortly on the reset button (located on the front panel); the gateway resets a second time and is restored with factory default parameters (username: "Admin", password: "Admin").
4. Reconnect the network cable.
5. Assign the Mediant 2000 IP address (refer to Section 4.1 on page 35).
6. Load your previously backed-up *ini* file, or the default *ini* file (received with the software kit). To load the *ini* file via the Embedded Web Server, refer to Section 5.9.5 on page 69.

## 4.3 Configuring the Mediant 2000 *Basic* Parameters

To configure the Mediant 2000 *basic* parameters use the Embedded Web Server's 'Quick Setup' screen (shown in Figure 4-1 below). Refer to Section 5.6 on page 41 for information on accessing the 'Quick Setup' screen.

Figure 4-1: Mediant 2000 Quick Setup Screen

Quick Setup	
<b>IP Configuration</b>	
IP Address	10.8.25.123
NAT IP Address	0.0.0.0
Subnet Mask	255.255.0.0
Default Gateway IP Address	10.8.0.1
<b>SIP Parameters</b>	
Gateway Name	10.8.8.10
Working with Proxy	No
Proxy IP Address	10.8.8.10
Proxy Name	
Enable Registration	No
<b>Coder Name</b> (msec)	
<input checked="" type="checkbox"/> 1st Coder	g711Alaw64k 20
<b>Tables</b>	
Tel to IP Routing Table	-->
Trunk Group Table	-->

➤ **To configure basic SIP parameters, take these 10 steps:**

1. If the Mediant 2000 is connected to a router with NAT (Network Address Translation) enabled, perform the following procedure. If it isn't, leave the 'NAT IP Address' field undefined.
  - Determine the "public" IP address assigned to the router (by using, for instance, router Web management). Enter this public IP address in the 'NAT IP Address' field.

- Enable the DMZ (Demilitarized Zone) configuration on the residential router for the LAN port where the Mediant 2000 gateway is connected. This enables unknown packets to be routed to the DMZ port.
- 2. Under 'SIP Parameters', enter the Mediant 2000 domain name in the field 'Gateway Name'. If the field is not specified, the Mediant 2000 IP address is used instead (default).
- 3. When working with a Proxy server, set 'Working with Proxy' field to 'Yes' and enter the IP address of the primary Proxy server in the field 'Proxy IP address'. When no Proxy is used, the internal routing table is used to route the calls.
- 4. Enter the Proxy name in the field 'Proxy Name'. If Proxy name is used, it replaces the Proxy IP address in all SIP messages. This means that messages is still sent to the physical Proxy IP address but the SIP URI contains the Proxy name instead.
- 5. Configure 'Enable Registration' to 'Yes' or 'No':  
 'No' = the Mediant 2000 does not register to a Proxy server/Registrar (default).  
 'Yes' = the Mediant 2000 registers to a Proxy server/Registrar at power up and every 'Registration Time' seconds. For detailed information on the parameter 'Registration Time', refer to [Table 6-3](#) on page 100.
- 6. Select the coder (i.e., vocoder) that best suits your VoIP system requirements. The default coder is: G.7231 30 msec. To program the entire list of coders you want the Mediant 2000 to use, click the button on the left side of the '1st Coder' field; the drop-down lists for the 2nd to 5th coders appear. Select coders according to your system requirements. Note that coders higher on the list are preferred and take precedence over coders lower on the list.



**Note:** The preferred coder is the coder that the Mediant 2000 uses as a first choice for all connections. If the far end gateway does not use this coder, the Mediant 2000 negotiates with the far end gateway to select a coder that both sides can use.

- 7. To program the Tel to IP Routing Table, press the arrow button next to 'Tel to IP Routing Table'. For information on how to configure the Tel to IP Routing Table, refer to [Section 5.8.4.1](#) on page 49.
- 8. To program the E1/T1 B-channels, press the arrow button next to 'Trunk Group Table'. For information on how to configure the Trunk Group Table, refer to [Section 5.8.6](#) on page 58.
- 9. Click the Reset button and click OK in the prompt; The Mediant 2000 applies the changes and restarts, taking approximately 3 minutes to complete. When the Mediant 2000 has finished restarting, the Ready and LAN LEDs on the front panel are lit green.
- 10. After the gateway was reset, access the Advanced Configuration>Trunk Settings page, and select the gateway's E1/T1 protocol type and Framing method that best suits your system requirements. Note that for E1 spans, the framing method must always be set to 'Extended Super Frame'. For information on how to configure the Trunk Settings, refer to [Section 5.9.3](#) on page 66.

You are now ready to start using the gateway. To prevent unauthorized access to the Mediant 2000, it is recommended that you change the username and password that are used to access the Web Interface. Refer to [Section 5.9.7](#) on page 71 for details on how to change the username and password.



**Tip:** Once the gateway is configured correctly back up your settings by making a copy of the VoIP gateway configuration (*ini* file) and store it in a directory on your PC. This saved file can be used to restore configuration settings at a future time. For information on backing up and restoring the gateway's configuration, refer to [Section 5.9.5](#) on page 69.

## 5 Web Management

### 5.1 Configuration Concepts

Users can utilize the Mediant 2000 in a wide variety of applications, enabled by its parameters and configuration files (e.g., Call Progress Tones (CPT), etc.). The parameters can be configured and configuration files can be loaded using:

- A standard Web Browser (described and explained in this section).
- A configuration file referred to as the *ini* file. For information on how to use the *ini* file, refer to Section 6 on page 87.
- An SNMP browser software (refer to Section 11 on page 171).
- AudioCodes' Element Management System (EMS) (refer to Section 11.8 on page 182 and to AudioCodes' EMS User's Manual or EMS Product Description).

To upgrade the Mediant 2000 (load new software or configuration files onto the gateway) use the Software Upgrade wizard, available through the Web Interface (refer to Section 5.11.1 on page 78), or alternatively use the BootP/TFTP configuration utility (refer to Section 10.3.1 on page 169).

For information on the configuration files, refer to Section 7 on page 135.

### 5.2 Overview of the Embedded Web Server

The Embedded Web Server is used both for gateway configuration, including loading of configuration files, and for run-time monitoring. The Embedded Web Server can be accessed from a standard Web browser, such as Microsoft™ Internet Explorer, Netscape™ Navigator, etc. Specifically, Users can employ this facility to set up the gateway configuration parameters. Users also have the option to remotely reset the gateway and to permanently apply the new set of parameters.

### 5.3 Computer Requirements

To use the Web Interface, the following is needed:

- A computer capable of running your Web browser.
- A network connection to the VoIP gateway.
- One of the following compatible Web browsers:
  - Microsoft™ Internet Explorer™ (version 6.0 and higher).
  - Netscape™ Navigator™ (version 7.0 and higher).



**Note:** Some Java-script applications are not supported in Netscape.

## 5.4 Password Control

The Embedded Web Server is protected by a unique username and password combination. The first time a browser request is made, the User is requested to provide his username and password to obtain access. Subsequent requests are negotiated by the browser on behalf of the User, so that the User doesn't have to re-enter the username and password for each request, but the request is still authenticated (the Embedded Web Server uses the MD5 authentication method supported by the HTTP 1.1 protocol).

An additional level of protection is obtained by a restriction that no more than three IP addresses can access the Embedded Web Server concurrently. With this approach, a fourth User is told that the server is busy, even if the correct username and password were provided.

### 5.4.1 Embedded Web Server Username & Password

The default username and password for all gateways are:

- Username = "Admin" (case-sensitive)
- Password = "Admin" (case-sensitive)

For details on changing the username and password, refer to Section 5.9.7 on page 71. Note that the password and username can be a maximum of 7 case-sensitive characters.

The User can reset the Web username and password (to the default values) by enabling an *ini* file parameter called 'ResetWebPassword'. The Web password is automatically the default password.

## 5.5 Configuring the Web Interface via the *ini* File

Two additional security preferences can be configured using *ini* file parameters. These security levels provide protection against unauthorized access (such as Internet hacker attacks), particularly to Users without a firewall. For information on the *ini* file, refer to Section 6 on page 87.

### 5.5.1 Limiting the Embedded Web Server to Read-Only Mode

Users can limit the Web Interface to read-only mode by changing the *ini* file parameter 'DisableWebConfig' to 1. In this mode all Web screens are read-only and cannot be modified. In addition, the following screens cannot be accessed: 'Quick Setup', 'Change Password', 'Reset', 'Save Configuration', 'Software Upgrade Wizard', 'Load Auxiliary Files', 'Configuration File' and 'Regional Settings'.

### 5.5.2 Disabling the Embedded Web Server

To deny access to the gateway through HTTP protocol, the User can disable the Embedded Web Server task. To disable the Web task, use the *ini* file parameter 'DisableWebTask = 1'. The default is to Web task enabled.

## 5.6 Accessing the Embedded Web Server

➤ **To access the Embedded Web Server, take these 4 steps:**

1. Open a standard Web-browsing application such as Microsoft™ Internet Explorer™ (Version 6.0 and higher) or Netscape™ Navigator™ (Version 7.0 and higher).
2. In the Uniform Resource Locator (URL) field, specify the IP address of the Mediant 2000 (e.g., http://10.1.10.10); the Embedded Web Server's 'Enter Network Password' screen appears, shown in [Figure 5-1](#).

**Figure 5-1: Embedded Web Server Login Screen**



3. In the 'User Name' and 'Password' fields, enter the username (default: "Admin") and password (default: "Admin"). Note that the username and password are case-sensitive.
4. Click the **OK** button; the 'Quick Setup' screen is accessed (shown in [Figure 4-1](#)).

### 5.6.1 Using Internet Explorer to Access the Embedded Web Server

Internet explorer's security settings may block access to the gateway's Web browser if they're configured incorrectly. In this case, the following message is displayed:

**Unauthorized**

Correct authorization is required for this area. Either your browser does not perform authorization or your authorization has failed. RomPager server.

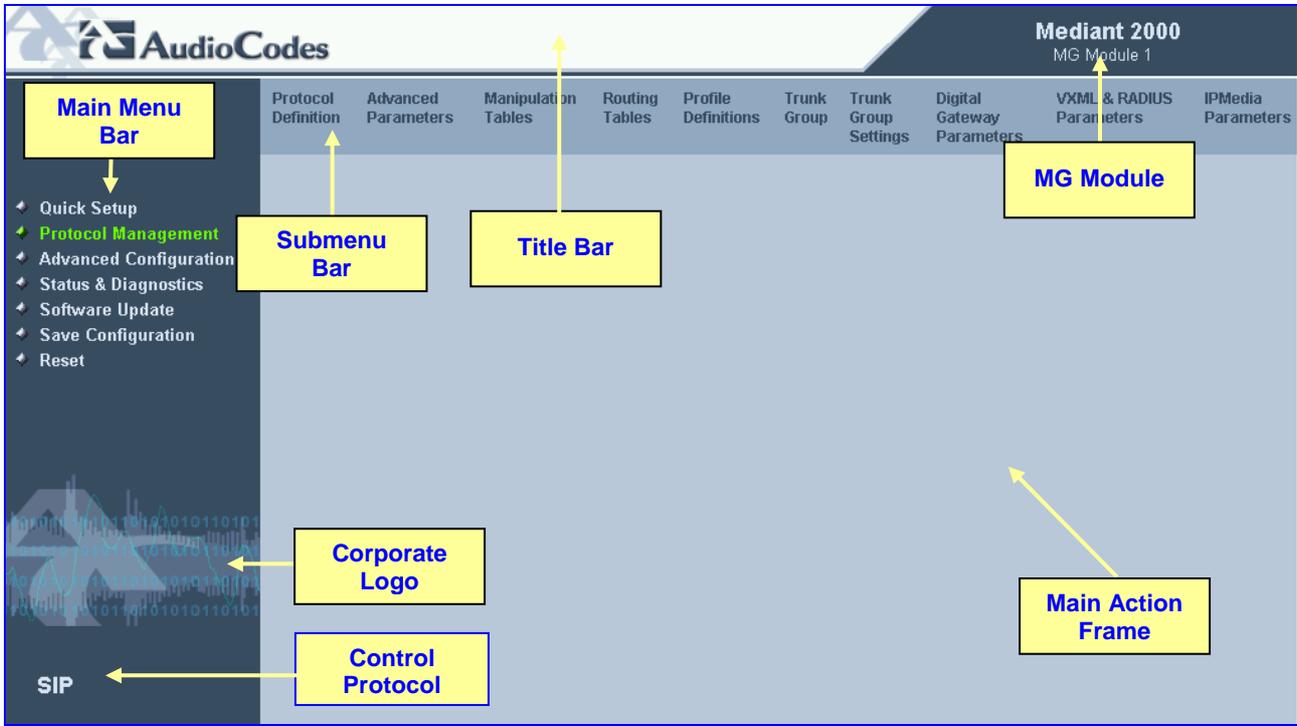
➤ **To troubleshoot blocked access to Internet Explorer, take these 2 steps:**

1. Delete all cookies from the Temporary Internet files. If this does not clear up the problem, the security settings may need to be altered (refer to Step 2).
2. In Internet Explorer, Tools, Internet Options select the Security tab, and then select Custom Level. Scroll down until the Logon options are displayed and change the setting to Prompt for username and password and then restart the browser. This fixes any issues related to domain use logon policy.

## 5.7 Getting Acquainted with the Web Interface

Figure 5-2 shows the general layout of the Web Interface screen.

Figure 5-2: Mediant 2000 Web Interface



The Web Interface screen features the following components:

- **Title bar** - contains three configurable elements: corporate logo, a background image and the product's name. For information on how to modify these elements, refer to [Appendix F](#) on page 207.
- **Main menu bar** - always appears on the left of every screen to quickly access parameters, submenus, submenu options, functions and operations.
- **Submenu bar** - appears on the top of screens and contains submenu options.
- **Main action frame** - the main area of the screen in which information is viewed and configured.
- **Corporate logo** – AudioCodes' corporate logo. For information on how to remove this logo, refer to [Appendix F](#) on page 207.
- **Control Protocol** – the Mediant 2000 control protocol.
- **MG Module** – the Mediant 2000 media gateway module (Module 1 or Module 2).

### 5.7.1 Main Menu Bar

The main menu bar of the Web Interface is divided into the following 7 menus:

- **Quick Setup** – Use this menu to configure the gateway's basic settings; for the full list of configurable parameters go directly to 'Protocol Management' and 'Advanced Configuration' menus. An example of the Quick Setup configuration is described in Section 4.2 on page 36.
- **Protocol Management** – Use this menu to configure the gateway's control protocol parameters and tables (refer to Section 5.8 on page 44).

- **Advanced Configuration** – Use this menu to set the gateway's advanced configuration parameters (for advanced users only) (refer to Section 5.9 on page 62).
- **Status & Diagnostics** – Use this menu to view and monitor the gateway's channels, Syslog messages, hardware / software product information, and to assess the gateway's statistics and IP connectivity information (refer to Section 5.10 on page 71).
- **Software Update** – Use this menu when you want to load new software or configuration files onto the gateway (refer to Section 5.11 on page 78).
- **Save Configuration** – Use this menu to save configuration changes to the non-volatile flash memory (refer to Section 5.12 on page 84).
- **Reset** – Use this menu to remotely reset the gateway. Note that you can choose to save the gateway configuration to flash memory before reset (refer to Section 5.12 on page 84).

When positioning your cursor over a parameter name (or a table) for more than 1 second, a short description of this parameter is displayed. Note that those parameters that are preceded with an exclamation mark (!) are *Not* changeable on-the-fly and require reset.

### 5.7.2 Saving Changes

To save changes to the volatile memory (RAM) press the **Submit** button (changes to parameters with on-the-fly capabilities are immediately available, other parameter are updated only after a gateway reset). Parameters that are only saved to the volatile memory revert to their previous settings after hardware reset. When performing a software reset (i.e., via Web or SNMP) you can choose to save the changes to the non-volatile memory. To save changes so they are available after a power fail, you must save the changes to the non-volatile memory (flash). When **Save Configuration** is performed, all parameters are saved to the flash memory.

➤ **To save the changes to flash, take these 2 steps:**

1. Click the **Save Configuration** button; the 'Save Configuration to Flash Memory' screen appears.
2. Click the **Save Configuration** button in the middle of the screen; a confirmation message appears when the save is complete.

**Note:** When you reset the Mediant 2000 from the Web Interface, you can choose to save the configuration to flash memory.

### 5.7.3 Entering Phone Numbers in Various Tables

Phone numbers entered into various tables on the gateway, such as the Tel to IP routing table, must be entered without any formatting characters. For example, if you wish to enter the phone number 555-1212, it must be entered as 5551212 without the hyphen (-). If the hyphen is entered, the entry does not work. The hyphen character is used in number entry only, as part of a range definition. For example, the entry [20-29] means "all numbers in the range 20 to 29".

## 5.8 Protocol Management

Use this menu to configure the gateway's SIP parameters and tables.

### 5.8.1 Protocol Definition Parameters

Use this submenu to configure the following gateway's specific SIP protocol parameters:

- General Parameters
- Proxy & Registration Parameters
- Coders (refer to Section 5.8.1.1 below)
- DTMF & Dialing Parameters

#### 5.8.1.1 Coders

From the Coders screen you can configure the first to fifth preferred coders (and their corresponding ptime) for the gateway. The first coder is the highest priority coder and is used by the gateway whenever possible. If the far end gateway cannot use the coder assigned as the first coder, the gateway attempts to use the next coder and so forth.

➤ **To configure the gateway's coders, take these 6 steps:**

1. Open the 'Coders' screen (**Protocol Management** menu > **Protocol Definition** submenu > **Coders** option); the 'Coders' screen is displayed.

**Figure 5-3: Coders Screen**

Coders		
1st Coder	g711Ulaw64k	20
2nd Coder	g729	30
3rd Coder	g726	10
4th Coder		
5th Coder		

2. From the coder drop-down list, select the coder you want to use. For the full list of available coders and their corresponding ptime, refer to the *ini* file parameter 'CoderName' (described in Table 6-3).  
**Note:** Each coder can appear only once.
3. From the drop-down list to the right of the coder list, select the size of the Voice Packet (ptime) used with this coder in milliseconds. Selecting the size of the packet determines how many coder payloads are combined into one RTP (Real-Time Transport Protocol) (voice) packet.  
**Note 1:** The ptime packetization period depends on the selected coder name.  
**Note 2:** If not specified, the ptime gets a default value.  
**Note 3:** The ptime specifies the maximum packetization time the gateway can receive.
4. Repeat steps 2 and 3 for the second to fifth coders (optional).
5. Click the **Submit** button to save your changes.
6. To save the changes so they are available after a power fail, refer to Section 5.12 on page 84.



**Note:** Only the ptime of the first coder in the defined coder list is declared in Invite/200 OK SDP, even if multiple coders are defined.

## 5.8.2 Advanced Parameters

Use this submenu to configure the following gateway's advanced control protocol parameters.

- Disconnect and Answer Supervision
- CDR and Debug
- Miscellaneous Parameters
- Supplementary Services

## 5.8.3 Number Manipulation Tables

The VoIP gateway provides four Number Manipulation tables for incoming and outgoing calls. These tables are used to modify the destination and source telephone numbers so that the calls can be routed correctly.

The Manipulation Tables are:

- Destination Phone Number Manipulation Table for IP→Tel calls
- Destination Phone Number Manipulation Table for Tel→IP call
- Source Phone Number Manipulation Table for IP→Tel calls
- Source Phone Number Manipulation Table for Tel→IP calls



**Note:** Number manipulation can be performed either before or after a routing decision is made. For example, you can route a call to a specific trunk group according to its original number, and then you can remove/add a prefix to that number before it is routed. To control when number manipulation is done, set the 'RouteModelP2Tel' and the 'RouteModeTel2IP' parameters. For information on these parameters, refer to [Table 6-5](#) on page [115](#).

Possible uses for number manipulation can be as follows:

- To strip/add dialing plan digits from/to the number. For example, a user could dial 9 in front of each number to indicate an external line. This number (9) can be removed here before (after) the call is setup.
- Assignment of NPI/TON to IP→Tel calls. The VoIP gateway can use a single global setting for NPI/TON classification or it can use the setting in this table on a call by call basis. Control for this is done using "Protocol Management>Protocol Definition>Destination/Source Number Encoding Type".
- Allow / disallow Caller ID information to be sent according to destination / source prefixes.

### ➤ To configure the Number Manipulation tables, take these 5 steps:

1. Open the Number Manipulation screen you want to configure (**Protocol Management** menu > **Manipulation Tables** submenu); the relevant Manipulation table screen is displayed. [Figure 5-4](#) shows the 'Source Phone Number Manipulation Table for Tel→IP calls'.

**Figure 5-4: Source Phone Number Manipulation Table for Tel→IP Calls**

	Dest. Prefix	Source Prefix	Num of Stripped Digits	Prefix (Suffix) to Add	Number of Digits to Leave	Presentation
1	03	201	0	972		Allowed
2		1001	4	5(23)		Restricted
3		123451001#	0	(8)	4	Not Configured
4		[30-40]xx	(1)	2		Not Configured
5	[6,7,8]	2001	5	3		Not Configured
6						Not Configured
7						Not Configured
8						Not Configured
9						Not Configured
10						Not Configured

2. In the 'Table Index' drop-down list, select the range of entries that you want to edit (up to 20 entries can be configured for Source Number Manipulation and 50 entries for Destination Number Manipulation).
3. Configure the Number Manipulation table according to [Table 5-1](#).
4. Click the **Submit** button to save your changes.
5. To save the changes so they are available after a power fail, refer to [Section 5.12](#) on page 84.

**Table 5-1: Number Manipulation Parameters**

Parameter	Description
Destination Prefix	Each entry in the Destination Prefix fields represents a destination telephone number prefix. An asterisk (*) represents any number.
Source Prefix	Each entry in the Source Prefix fields represents a source telephone number prefix. An asterisk (*) represents any number.
Source IP	Each entry in the Source IP fields represents the source IP address of the call (obtained from the Contact header in the Invite message). This column only applies to the 'Destination Phone Number Manipulation Table for Tel'. <b>Note:</b> The source IP address can include the "x" wildcard to represent <u>single</u> digits. For example: 10.8.8.xx represents all the addresses between 10.8.8.10 to 10.8.8.99.
The manipulation rules are applied to any incoming call whose: <ul style="list-style-type: none"> <li>• Destination number prefix matches the prefix defined in the 'Destination Number' field.</li> <li>• Source number prefix matches the prefix defined in the 'Source Prefix' field.</li> <li>• Source IP address matches the IP address defined in the 'Source IP' field (if applicable).</li> </ul> Note that number manipulation can be performed using a combination of each of the above criteria, or using each criterion independently. <b>Note:</b> For available notations that represent multiple numbers, refer to <a href="#">Section 5.8.3.1</a> on page 47.	
Num of stripped digits	<ul style="list-style-type: none"> <li>• Enter the number of digits that you want to remove from the left of the telephone number prefix. For example, if you enter 3 and the phone number is 5551234, the new phone number is 1234.</li> <li>• Enter the number of digits (in brackets) that you want to remove from the right of the telephone number prefix.</li> </ul> <b>Note:</b> A combination of the two options is allowed (e.g., 2(3)).

**Table 5-1: Number Manipulation Parameters**

Parameter	Description
Prefix / Suffix to add	<ul style="list-style-type: none"> <li>Prefix - Enter the number / string you want to add to the front of the phone number. For example, if you enter 9 and the phone number is 1234, the new number is 91234.</li> <li>Suffix - Enter the number / string (in brackets) you want to add to the end of the phone number. For example, if you enter (00) and the phone number is 1234, the new number is 123400.</li> </ul> <p><b>Note:</b> You can enter a prefix and a suffix in the same field (e.g., 9(00)).</p>
Number of digits to leave	Enter the number of digits that you want to leave from the right.
<p><b>Note:</b> The manipulation rules are executed in the following order:</p> <ol style="list-style-type: none"> <li>1. Num of stripped digits</li> <li>2. Number of digits to leave</li> <li>3. Prefix / suffix to add</li> </ol> <p>Figure 5-4 on the previous page exemplifies the use of these manipulation rules in the 'Source Phone Number Manipulation Table for Tel→IP Calls':</p> <ul style="list-style-type: none"> <li>• When destination number equals 035000 and source number equals 20155, the source number is changed to 97220155.</li> <li>• When source number equals 1001876, it is changed to 587623.</li> <li>• Source number 1234510012001 is changed to 20018.</li> <li>• Source number 3122 is changed to 2312.</li> </ul>	
NPI	<p>Select the Number Plan assigned to this entry. You can select Unknown <b>[0]</b>, Private <b>[9]</b> or E.164 Public <b>[1]</b>. The default is Unknown. For a detailed list of the available NPI/TON values, refer to Section 5.8.3.2 on page 48.</p>
TON	<p>Select the Number Type assigned to this entry.</p> <ul style="list-style-type: none"> <li>• If you selected Unknown as the Number Plan, you can select Unknown <b>[0]</b>.</li> <li>• If you selected Private as the Number Plan, you can select Unknown <b>[0]</b>, Level 2 Regional <b>[1]</b>, Level 1 Regional <b>[2]</b>, PSTN Specific <b>[3]</b> or Level 0 Regional (Local) <b>[4]</b>.</li> <li>• If you selected E.164 Public as the Number Plan, you can select Unknown <b>[0]</b>, International <b>[1]</b>, National <b>[2]</b>, Network Specific <b>[3]</b>, Subscriber <b>[4]</b> or Abbreviated <b>[6]</b>.</li> </ul> <p>The default is Unknown.</p>
Presentation	<p>Select 'Allowed' to send Caller ID information when a call is made using these destination / source prefixes. Select 'Restricted' if you want to restrict Caller ID information for these prefixes.</p>

### 5.8.3.1 Dialing Plan Notation

The dialing plan notation applies, in addition to the four Manipulation tables, also to Tel→IP Routing table and to IP→Trunk Group Routing table.

When entering a number in the destination and source 'Prefix' columns, you can create an entry that represents multiple numbers using the following notation:

- [n-m] represents a range of numbers
- [n,m] represents multiple numbers. Note that this notation only supports single digit numbers.
- x represents any single digit
- # (that terminates the number) represents the end of a number
- A single asterisk (\*) represents any number

For example:

- [5551200-5551300]# represents all numbers from 5551200 to 5551300

- [2,3,4]xxx# represents four-digit numbers that start with 2, 3 or 4
- 54324 represents any number that starts with 54324
- 54324xx# represents a 7 digit number that starts with 54324
- 123[100-200]# represents all numbers from 123100 to 123200.

The VoIP gateway matches the rules starting at the top of the table. For this reason, enter more specific rules above more generic rules. For example, if you enter 551 in entry 1 and 55 in entry 2, the VoIP gateway applies rule 1 to numbers that starts with 551 and applies rule 2 to numbers that start with 550, 552, 553, 554, 555, 556, 557, 558 and 559. However if you enter 55 in entry 1 and 551 in entry 2, the VoIP gateway applies rule 1 to all numbers that start with 55 including numbers that start with 551.

### 5.8.3.2 Numbering Plans and Type of Number

Numbers are classified by their Numbering Plan Indication (NPI) and their Type of Number (TON). The Mediant 2000 supports all NPI/TON classifications used in the standard. The list of ISDN ETSI NPI/TON values is shown as follows:

**Table 5-2: NPI/TON Values for ISDN ETSI**

NPI	TON	Description
Unknown [0]	Unknown [0]	A valid classification, but one that has no information about the numbering plan.
E.164 Public [1]	Unknown [0]	A public number in E.164 format, but no information on what kind of E.164 number.
	International [1]	A public number in complete international E.164 format. For example: 16135551234
	National [2]	A public number in complete national E.164 format. For example: 6135551234
	Subscriber [4]	A public number in complete E.164 format representing a local subscriber. For example: 5551234
Private [9]	Unknown [0]	A private number, but with no further information about the numbering plan
	Level 2 Regional [1]	
	Level 1 Regional [2]	A private number with a location. For example: 3932200
	PISN Specific [3]	
	Level 0 Regional (local) [4]	A private local extension number. For example: 2200

For NI-2 and DMS-100 ISDN variants the valid combinations of TON and NPI for calling and called numbers are (Plan/Type):

- 0/0 - Unknown/Unknown
- 1/1 - International number in ISDN/Telephony numbering plan
- 1/2 - National number in ISDN/Telephony numbering plan
- 1/4 - Subscriber (local) number in ISDN/Telephony numbering plan
- 9/4 - Subscriber (local) number in Private numbering plan

## 5.8.4 Configuring the Routing Tables

Use this submenu to configure the gateway's IP→Tel and Tel→IP routing tables and their associated parameters.

### 5.8.4.1 Tel to IP Routing Table

The Tel to IP Routing Table is used to route incoming Tel calls to IP addresses. This routing table associates a called / calling telephone number's prefixes with a destination IP address or with an FQDN (Fully Qualified Domain Name). When a call is routed through the VoIP gateway (Proxy isn't used), the called and calling numbers are compared to the list of prefixes on the IP Routing Table (up to 50 prefixes can be configured); Calls that match these prefixes are sent to the corresponding IP address. If the number dialed does not match these prefixes, the call is not made.

When using a Proxy server, you do not need to configure the Telephone to IP Routing Table. However, if you want to use fallback routing when communication with Proxy is lost, or to use the 'Filter Calls to IP' and IP Security features, or to obtain different SIP URI host names (per called number), you need to configure the IP Routing Table.

Note that for the Tel to IP Routing table to take precedence over a Proxy for routing calls, set the parameter 'PreferRouteTable' to 1. The gateway checks the 'Destination IP Address' field in the 'Tel to IP Routing' table for a match with the outgoing call. Only if a match is not found, a Proxy is used.

Possible uses for Telephone to IP Routing can be as follows:

- Can fallback to internal routing table if there is no communication with the Proxy.
- Call Restriction – (when Proxy isn't used), reject all outgoing Tel→IP calls that are associated with the destination IP address: 0.0.0.0.
- IP Security – When the IP Security feature is enabled (SecureCallFromIP = 1), the VoIP gateway accepts only those IP→Tel calls with a source IP address identical to one of the IP addresses entered in the Telephone to IP Routing Table.
- Filter Calls to IP – When a Proxy is used, the gateway checks the Tel→IP routing table before a telephone number is routed to the Proxy. If the number is not allowed (number isn't listed or a Call Restriction routing rule was applied), the call is released.
- Always Use Routing Table – When this feature is enabled (AlwaysUseRouteTable = 1), even if a Proxy server is used, the SIP URI host name in the sent INVITE message is obtained from this table. Using this feature users are able to assign a different SIP URI host name for different called and/or calling numbers.
- Assign Profiles to destination address (also when a Proxy is used).
- Alternative Routing – (When Proxy isn't used) an alternative IP destination for telephone number prefixes is available. To associate an alternative IP address to called telephone number prefix, assign it with an additional entry (with a different IP address), or use an FQDN that resolves to two IP addresses. Call is sent to the alternative destination when one of the following occurs:
  - No ping to the initial destination is available, or when poor Quality of Service (QoS) (delay or packet loss, calculated according to previous calls) is detected, or when a DNS host name is not resolved. For detailed information on Alternative Routing, refer to Section 8.6 on page 146.
  - When a release reason that is defined in the 'Reasons for Alternative Tel to IP Routing' table is received. For detailed information on the 'Reasons for Alternative Routing Tables', refer to Section 5.8.4.4 on page 54.

Alternative routing (using this table) is commonly implemented when there is no response to an Invite message (after Invite retransmissions). The gateway then issues an internal 408 'No Response' implicit release reason. If this reason is included in the 'Reasons for

Alternative Routing' table, the gateway immediately initiates a call to the redundant destination using the next matched entry in the 'Tel to IP Routing' table. Note that if a domain name in this table is resolved to two IP addresses, the timeout for Invite retransmissions can be reduced by using the parameter 'Number of RTX Before Hotswap'.

**Note:** If the alternative routing destination is the gateway itself, the call can be configured to be routed back to PSTN. This feature is referred to as "PSTN Fallback", meaning that if sufficient voice quality is not available over the IP network, the call is routed through legacy telephony system (PSTN).



**Tip:** Tel to IP routing can be performed either before or after applying the number manipulation rules. To control when number manipulation is done, set the RouteModeTel2IP parameter. For information on this parameter, refer to refer to [Table 6-5](#) on page 115.

➤ **To configure the Tel to IP Routing table, take these 6 steps:**

1. Open the 'Tel to IP Routing' screen (**Protocol Management** menu > **Routing Tables** submenu > **Tel to IP Routing** option); the 'Tel to IP Routing' screen is displayed (shown in [Figure 5-5](#)).
2. In the 'Tel to IP Routing Mode' field, select the Tel to IP routing mode (refer to [Table 6-5](#)).
3. In the 'Routing Index' drop-down list, select the range of entries that you want to edit.
4. Configure the Tel to IP Routing table according to [Table 5-3](#).
5. Click the **Submit** button to save your changes.
6. To save the changes so they are available after a power fail, refer to [Section 5.12](#) on page 84.

**Figure 5-5: Tel to IP Routing Table Screen**

	Dest. Phone Prefix	Source Phone Prefix	Dest. IP Address	Profile ID	Status
1	10	100	10.33.45.63	1	OK
2	20	*	10.33.45.60	1	QOS Low
3	[3,4,6]	*	10.33.45.64	1	OK
4	54324	[1,2]	Domain.com	1	Dns Error
5	9	*	0.0.0.0	2	n/a
6	8xx#	*	10.13.77.7	1	Ping Error
7	*	*	10.13.77.7	1	OK
8					

**Table 5-3: Tel to IP Routing Table**

Parameter	Description
Destination Phone Prefix	Each entry in the Destination Phone Prefix fields represents a called telephone number prefix. The prefix can be 1 to 19 digits long. An asterisk (*) represents all numbers.
Source Phone Prefix	Each entry in the Source Phone Prefix fields represents a calling telephone number prefix. The prefix can be 1 to 19 digits long. An asterisk (*) represents all numbers.

Table 5-3: Tel to IP Routing Table

Parameter	Description
	<p>Any telephone number whose destination number matches the prefix defined in the 'Destination Phone Prefix' field <i>and</i> its source number matches the prefix defined in the adjacent 'Source Phone Prefix' field, is sent to the IP address entered in the 'IP Address' field.</p> <p>Note that Tel to IP routing can be performed according to a combination of source and destination phone prefixes, or using each independently.</p> <p><b>Note 1:</b> An additional entry of the same prefixes can be assigned to enable alternative routing.</p> <p><b>Note 2:</b> For available notations that represent multiple numbers, refer to Section 5.8.3.1 on page 47.</p>
Destination IP Address	<p>In each of the IP Address fields, enter the IP address that is assigned to these prefixes. Domain names, such as domain.com, can be used instead of IP addresses. To discard outgoing IP calls, enter 0.0.0.0 in this field.</p> <p><b>Note:</b> When using domain names, you must enter a DNS server IP address, or alternatively define these names in the 'Internal DNS Table'.</p>
Profile ID	Enter the number of the IP profile that is assigned to the destination IP address defined in the 'Destination IP Address' field.
Status	<p>A read only field representing the quality of service of the destination IP address.</p> <p>N/A = Alternative Routing feature is disabled.</p> <p>OK = IP route is available</p> <p>Ping Error = No ping to IP destination, route is not available</p> <p>QoS Low = Bad QoS of IP destination, route is not available</p> <p>DNS Error = No DNS resolution (only when domain name is used instead of an IP address).</p>

### 5.8.4.2 IP to Trunk Group Routing Table

The IP to Trunk Group Routing Table is used to route incoming IP calls to groups of E1/T1 B-channels called trunk groups. Calls are assigned to trunk groups according to any combination of the following three options (or using each independently):

- Destination phone prefix
- Source phone prefix
- Source IP address

The call is then sent to the VoIP gateway channels assigned to that trunk group. The specific channel, within a trunk group, that is assigned to accept the call is determined according to the trunk group's channel selection mode which is defined in the Trunk Group Settings table (Section 5.8.7 on page 60), or according to the global parameter 'ChannelSelectMode' (refer to Table 6-5 on page 115).

**Note:** When a release reason that is defined in the 'Reasons for Alternative IP to Tel Routing' table is received for a specific IP→Tel call, an alternative trunk group for that call is available. To associate an alternative trunk group to an incoming IP call, assign it with an additional entry in the 'IP to Trunk Group Routing' table (repeat the same routing rules with a different trunk group ID). For detailed information on the 'Reasons for Alternative Routing Tables', refer to Section 5.8.4.4 on page 54.

To use trunk groups you must also do the following:

- You must assign a trunk group ID to the VoIP gateway E1/T1 B-channels on the Trunk Group Table. For information on how to assign a trunk group ID to a B-channel, refer to Section 5.8.6 on page 58.

- You can configure the Trunk Group Settings table to determine the method in which new calls are assigned to channels within the trunk groups (a different method for each trunk group can be configured). For information on how to enable this option, refer to Section 5.8.7 on page 60. If a Channel Select Mode for a specific trunk group isn't specified, then the global 'Channel Select Mode' parameter (defined in 'General Parameters' screen under 'Advanced Parameters') applies.

➤ **To configure the IP to Trunk Group Routing table, take these 6 steps:**

- Open the 'IP to Trunk Group Routing' screen (**Protocol Management** menu > **Routing Tables** submenu > **IP to Trunk Group Routing** option); the 'IP to Trunk Group Routing' table screen is displayed.

**Figure 5-6: IP to Trunk Group Routing Table**

	Dest. Phone Prefix	Source Phone Prefix	Source IP Address	Trunk Group ID	Profile ID
1	10	*	0	1	2
2	20	101	0	1	2
3					
4					
5	[5010-5020]	*	0	3	1
6	6xx	*	0	3	1
7	71234#	*	0	3	1
8	*	*	0	4	3
9					
10					
11					
12					

- In the 'IP to Tel Routing Mode' field, select the IP to Tel routing mode (refer to Table 6-5 on page 115).
- In the 'Routing Index' drop-down list, select the range of entries that you want to edit (up to 24 entries can be configured).
- Configure the IP to Trunk Group Routing table according to Table 5-4.
- Click the **Submit** button to save your changes.
- To save the changes so they are available after a power fail, refer to Section 5.12 on page 84.

**Table 5-4: IP to Trunk Group Routing Table**

Parameter	Description
Destination Phone Prefix	Each entry in the Destination Phone Prefix fields represents a called telephone number prefix. The prefix can be 1 to 49 digits long. An asterisk (*) represents all numbers.
Source Phone Prefix	Each entry in the Source Phone Prefix fields represents a calling telephone number prefix. The prefix can be 1 to 49 digits long. An asterisk (*) represents all numbers.
Source IP Address	Each entry in the Source IP Address fields represents the source IP address of an IP→Tel call (obtained from the Contact header in the Invite message). <b>Note:</b> The source IP address can include the "x" wildcard to represent single digits. For example: 10.8.8.xx represents all the addresses between 10.8.8.10 to 10.8.8.99.

**Table 5-4: IP to Trunk Group Routing Table**

Parameter	Description
<p>Any SIP incoming call whose destination number matches the prefix defined in the 'Destination Phone Prefix' field <i>and</i> its source number matches the prefix defined in the adjacent 'Source Phone Prefix' field <i>and</i> its source IP address matches the address defined in the 'Source IP Address' field, is assigned to the trunk group entered in the field to the right of these fields.</p> <p>Note that IP to trunk group routing can be performed according to any combination of source / destination phone prefixes and source IP address, or using each independently.</p> <p><b>Note:</b> For available notations that represent multiple numbers (used in the prefix columns), refer to Section 5.8.3.1 on page 47.</p>	
Trunk Group ID	In each of the Trunk Group ID fields, enter the trunk group ID to which calls that match these prefixes are assigned.
Profile ID	Enter the number of the IP profile that is assigned to the routing rule.

### 5.8.4.3 Internal DNS Table

The internal DNS table, similar to a DNS resolution, translates hostnames into IP addresses. This table is used when hostname translation is required (e.g., 'Tel to IP Routing' table, etc.). Two different IP addresses can be assigned to the same hostname. If the hostname isn't found in this table, the gateway communicates with an external DNS server.

Assigning two IP addresses to hostname can be used for alternative routing (using the 'Tel to IP Routing' table).

➤ **To configure the internal DNS table, take these 7 steps:**

1. Open the 'Internal DNS Table' screen (**Protocol Management** menu > **Routing Tables** submenu > **Internal DNS Table** option); the 'Internal DNS Table' screen is displayed.

**Figure 5-7: Internal DNS Table Screen**

Internal DNS Table			
	DNS Name	First IP Address	Second IP Address
1	DomainName.com	10.8.21.4	10.13.2.95
2			
3			

2. In the 'DNS Name' field, enter the hostname to be translated. You can enter a string up to 31 characters long.
3. In the 'First IP Address' field, enter the first IP address that the hostname is translated to.
4. In the 'Second IP Address' field, enter the second IP address that the hostname is translated to.
5. Repeat steps 2 to 4, for each Internal DNS Table entry.
6. Click the **Submit** button to save your changes.
7. To save the changes so they are available after a power fail, refer to Section 5.12 on page 84.

### 5.8.4.4 Reasons for Alternative Routing

The Reasons for Alternative Routing screen includes two tables (Tel→IP and IP→Tel). Each table enables you to define up to 4 different release reasons. If a call is released as a result of one of these reasons, the gateway tries to find an alternative route to that call. The release reason for IP→Tel calls is provided in Q.931 notation. The release reason for Tel→IP calls is provided in SIP 4xx, 5xx and 6xx response codes. For Tel→IP calls an alternative IP address, for IP→Tel calls an alternative trunk group.

Refer to 'Tel to IP Routing Table' on page 49 for information on defining an alternative IP address. Refer to the 'IP to Trunk Group Routing Table' on page 51 for information on defining an alternative trunk group.

**You can use this table for example:**

For Tel→IP calls, when there is no response to an Invite message (after Invite retransmissions), and the gateway then issues an internal 408 'No Response' implicit release reason.

For IP→Tel calls, when the destination is busy, and release reason #17 is issued or for other call releases that issue the default release reason (#3). Refer to 'DefaultReleaseCause' in Table 6-3.

**Note:** The reasons for alternative routing option for Tel→IP calls only applies when Proxy isn't used.

➤ **To configure the reasons for alternative routing, take these 5 steps:**

1. Open the 'Reasons for Alternative Routing' screen (**Protocol Management** menu > **Routing Tables** submenu > **Reasons for Alternative Routing** option); the 'Reasons for Alternative Routing' screen is displayed.

**Figure 5-8: Reasons for Alternative Routing Screen**

Reasons for Redundant Routing	
<b>IP to Tel Reasons</b>	
Reason 1	3
Reason 2	17
Reason 3	6
Reason 4	1
<b>Tel to IP Reasons</b>	
Reason 1	408
Reason 2	486
Reason 3	
Reason 4	

2. In the 'IP to Tel Reasons' table, from the drop-down list select up to 4 different call failure reasons that invoke an alternative IP to Tel routing.
3. In the 'Tel to IP Reasons' table, from the drop-down list select up to 4 different call failure reasons that invoke an alternative Tel to IP routing.
4. Click the **Submit** button to save your changes.
5. To save the changes so they are available after a power fail, refer to Section 5.12 on page 84.

## 5.8.5 Configuring the Profile Definitions

Utilizing the Profiles feature, the Mediant 2000 provides high-level adaptation when connected to a variety of equipment (from both Tel and IP sides) and protocols, each of which require a different system behavior. Using Profiles, users can assign different Profiles (behavior) on a per-call basis, using the Tel to IP and IP to Trunk Group Routing tables, or associate different Profiles to the gateway's B-channels(s). The Profiles contain parameters such as Coders, T.38 Relay, Voice and DTMF Gains, Silence Suppression, Echo Canceler, RTP DiffServ and more. The Profiles feature allows users to tune these parameters or turn them on or off, per source or destination routing and/or the specific gateway or its ports. For example, specific E1/T1 spans can be designated for to have a profile which always uses G.711.

Each call can be associated with one or two Profiles, Tel Profile and (or) IP Profile. If both IP and Tel profiles apply to the same call, the coders and other common parameters of the preferred Profile (determined by the Preference option) are applied to that call. If the Preference of the Tel and IP Profiles is identical, the Tel Profile parameters are applied.



**Note:** The default values of the parameters in the Tel and IP Profiles are identical to the *Web/ini* file parameter values. If a value of a parameter is changed in the *Web/ini* file, it is automatically updated in the Profiles correspondingly. After any parameter in the Profile is modified by the user, modifications to parameters in the *Web/ini* file no longer impact that Profile.

### 5.8.5.1 Coder Group Settings

Use the Coders Group Settings screen to define up to four different coder groups. These coder groups are used in the Tel and IP Profile Settings screens to assign different coders to Profiles.

➤ **To configure the coder group settings, take these 8 steps:**

1. Open the 'Coder Group Settings' screen (**Protocol Management** menu > **Profile Definitions** submenu > **Coder Group Settings** option); the 'Coder Group Settings' screen is displayed.

**Figure 5-9: Coder Group Settings Screen**

Coder Group Settings		
Coder Group ID	1	
1st Coder	g711Alaw64k	20
2nd Coder	g711Ulaw64k	10
3rd Coder		20
4th Coder		20
5th Coder		20

2. In the 'Coder Group ID' drop-down list, select the coder group you want to edit (up to four coder groups can be configured).
3. From the coder drop-down list, select the coder you want to use. For the full list of available coders and their corresponding ptime, refer to the *ini* file parameter 'CoderName\_ID' (described in [Table 6-3](#)).  
**Note:** Each coder can appear only once.
4. From the drop-down list to the right of the coder list, select the size of the Voice Packet (ptime) used with this coder in milliseconds. Selecting the size of the packet determines how many coder payloads are combined into one RTP (voice) packet.  
**Note 1:** The ptime packetization period depends on the selected coder name.

**Note 2:** If not specified, theptime gets a default value.

**Note 3:** Theptime specifies the maximum packetization time the gateway can receive.

5. Repeat steps 3 and 4 for the second to fifth coders (optional).
6. Repeat steps 2 to 5 for the second to forth coder groups (optional).
7. Click the **Submit** button to save your changes.
8. To save the changes so they are available after a power fail, refer to Section 5.12 on page 84.



**Note:** In the current version, only theptime of the first coder is sent in the SDP section of the Invite message.

### 5.8.5.2 Tel Profile Settings

Use the Tel Profile Settings screen to define up to four different Tel Profiles. These Profiles are used in the 'Trunk Group' table to associate different Profiles to gateway's B-channels, thereby applying different behavior to different Mediant 2000 B-channels.

➤ **To configure the Tel Profile settings, take these 8 steps:**

1. Open the 'Tel Profile Settings' screen (**Protocol Management** menu > **Profile Definitions** submenu > **Tel Profile Settings** option); the 'Tel Profile Settings' screen is displayed.

**Figure 5-10: Tel Profile Settings Screen**

Tel Profile Settings	
Profile ID	1
Profile Name	Default Tel Profile
Profile Parameters	
Profile Preference	1
Fax Signaling Method	No Fax
Dynamic Jitter Buffer Minimum Delay [msec]	70
Dynamic Jitter Buffer Optimization Factor	7
RTP IP Diff Serv	0
Signaling DiffServ	0
Voice Volume (-32 to 31 dB)	0
DTMF Volume (-31 to 0 dB)	-11
Input Gain (-32 to 31 dB)	0
Echo Canceler	Enable
Coder Group	
Coder Group	Default Coder Group

2. In the 'Profile ID' drop-down list, select the Tel Profile you want to edit (up to four Tel Profiles can be configured).

3. In the 'Profile Preference' drop-down list, select the preference (1-10) of the current Profile. The preference option is used to determine the priority of the Profile. If both IP and Tel profiles apply to the same call, the coders and other common parameters of the preferred Profile are applied to that call. If the Preference of the Tel and IP Profiles is identical, the Tel Profile parameters are applied.  
**Note:** If the coder lists of both IP and Tel Profiles apply to the same call, an intersection of the coders is performed (i.e., only common coders remain). The order of the coders is determined by the preference.
4. Configure the Profile's parameters according to your requirements. For detailed information on each parameter refer to the description of the screen in which it is configured as an individual parameter.
5. In the 'Coder Group' drop-down list, select the coder group you want to assign to that Profile. You can select the gateway's default coders (refer to Section 5.8.1.1 on page 44) or one of the coder groups you defined in the Coder Group Settings screen (refer to Section 5.8.5.1 on page 55).
6. Repeat steps 2 to 6 for the second to fifth Tel Profiles (optional).
7. Click the **Submit** button to save your changes.
8. To save the changes so they are available after a power fail, refer to Section 5.12 on page 84.

### 5.8.5.3 IP Profile Settings

Use the IP Profile Settings screen to define up to four different IP Profiles. These Profiles are used in the Tel to IP and IP to Trunk Group Routing tables to associate different Profiles to routing rules. IP Profiles can also be used when working with Proxy server (set 'AlwaysUseRouteTable' to 1).

➤ **To configure the IP Profile settings, take these 8 steps:**

1. Open the 'IP Profile Settings' screen (**Protocol Management** menu > **Profile Definitions** submenu > **IP Profile Settings** option); the 'IP Profile Settings' screen is displayed.

**Figure 5-11: IP Profile Settings Screen**

IP Profile Settings	
Profile ID	1
Profile Name	Default Ip Profile
Profile Parameters	
Profile Preference	1
Fax Signaling Method	No Fax
Dynamic Jitter Buffer Minimum Delay [msec]	70
Dynamic Jitter Buffer Optimization Factor	7
RTP IP Diff Serv	0
Signaling DiffServ	0
Silence Suppression	Disable
RTP Redundancy Depth	0
Remote RTP Base UDP Port	0
Coder Group	
Coder Group	Default Coder Group

2. In the 'Profile ID' drop-down list, select the IP Profile you want to edit (up to four IP Profiles can be configured).
3. In the 'Profile Preference' drop-down list, select the preference (1-10) of the current Profile. The preference option is used to determine the priority of the Profile. If both IP and Tel profiles apply to the same call, the coders and other common parameters of the preferred Profile are applied to that call. If the Preference of the Tel and IP Profiles is identical, the Tel Profile parameters are applied.  
**Note:** If the coder lists of both IP and Tel Profiles apply to the same call, an intersection of the coders is performed (i.e., only common coders remain). The order of the coders is determined by the preference.
4. Configure the Profile's parameters according to your requirements. For detailed information on each parameter refer to the description of the screen in which it is configured as an individual parameter.
5. In the 'Coder Group' drop-down list, select the coder group you want to assign to that Profile. You can select the gateway's default coders (refer to Section 5.8.1.1 on page 44) or one of the coder groups you defined in the Coder Group Settings screen (refer to Section 5.8.5.1 on page 55).
6. Repeat steps 2 to 6 for the second to fifth IP Profiles (optional).
7. Click the **Submit** button to save your changes.
8. To save the changes so they are available after a power fail, refer to Section 5.12 on page 84.

### 5.8.6 Configuring the Trunk Group Table

Use the Trunk Group table to assign trunk groups, profiles and logical telephone numbers to the gateway's E1/T1 B-channels. Trunk Groups are used for routing IP→Tel calls with common rules. Channels that are not defined are disabled.

➤ **To configure the Trunk Group table, take these 4 steps:**

1. Open the 'Trunk Group Table' screen (**Protocol Management** menu > **Trunk Group**); the 'Trunk Group Table' screen is displayed.

**Figure 5-12: Trunk Group Table Screen**

	Trunk ID	Channels	Phone Number	Trunk Group ID	Profile ID
1	1	1-24	4000	1	2
2	2	1-12	5000	2	2
3	2	13-24	6000	3	1
4					
5					
6					

2. Configure the Trunk Group according to Table 5-5
3. Click the **Submit** button to save your changes.
4. To save the changes so they are available after a power fail, refer to Section 5.12 on page 84.

Table 5-5: Trunk Group Table

Parameter	Description
Trunk ID	The numbers (1-8) in the Trunk ID drop-down list represent the physical trunks on the back of the VoIP gateway.
Channels	To enable the trunk's B-channels, you <b>must</b> enter their number in this field. [n-m] represents a range of channels. For example, enter [1-24] to specify the channels from 1 to 24. <b>Note:</b> The number of defined channels must not exceed the number of the trunk's B-channels (1-24 for T1 spans and 1-30 for E1 spans).
Phone Number	In each of the Phone Number fields, enter the first number in an ordered sequence that is assigned to the range of channels defined in the adjacent 'Channels' field. <b>Note:</b> This field is optional. The logical numbers defined in this field are used when an incoming PSTN / PBX call doesn't contain the calling number or called number (the latter being determined by the parameter 'ReplaceEmptyDstWithPortNumber'), these numbers are used to replace them. These logical numbers are also used for B-channel allocation for IP to Tel calls, if the trunk group's 'Channel Select Mode' is set to 'By Phone Number'.
Trunk Group ID	In each of the Trunk Group ID fields, enter the trunk group ID (1-99) assigned to the channels. The same trunk group ID can be used for more than one group of channels.  Trunk group ID is used to define a group of common behavior channels that are used for routing IP to Tel calls. If an IP to Tel call is assigned to a trunk group, the call is routed to the channel or channels that correspond to the trunk group ID.  You can configure the Trunk Group Settings table to determine the method in which new calls are assigned to channels within the trunk groups (refer to Section 5.8.7 on page 60).  <b>Note:</b> You must configure the IP to Trunk Group Routing Table (assigns incoming IP calls to the appropriate trunk group). If you do not configure the IP to Trunk Group Routing Table, calls do not complete. For information on how to configure this table, refer to Section 5.8.4.2 on page 51.
Profile ID	Enter the number of the Tel profile that is assigned to the B-channels defined in the 'Channels' field.

### 5.8.7 Configuring the Trunk Group Settings

The Trunk Group Settings Table is used to determine the method in which new calls are assigned to B-channels within each trunk group. If such a rule doesn't exist (for a specific Trunk group), the global rule, defined by the Channel Select Mode parameter (Protocol Definition > General Parameters), applies.

➤ **To configure the Trunk Group Settings table, take these 7 steps:**

1. Open the 'Trunk Group Settings' screen (**Protocol Management** menu > **Trunk Group Settings**); the 'Trunk Group Settings' screen is displayed.

**Figure 5-13: Trunk Group Settings Screen**

Trunk Group ID	Channel Select Mode
1	Cyclic Ascending
2	Ascending
3	Descending
4	
5	
6	
7	
8	
9	
10	
11	
12	

2. In the **Routing** Index drop-down list, select the range of entries that you want to edit (up to 24 entries can be configured).
3. In the **Trunk Group ID** field, enter the Trunk Group ID number.
4. In the **Channel Select Mode** drop-down list, select the Channel Select Mode that determines the method in which new calls are assigned to B-channels within the Trunk groups entered in the field to the right of this field. For information on available Channel Select Modes, refer to [Table 5-6](#).
5. Repeat steps 4 and 5, for each defined Trunk group.
6. Click the **Submit** button to save your changes.
7. To save the changes so they are available after a power fail, refer to [Section 5.12](#) on page 84.

**Table 5-6: Channel Select Modes**

Mode	Description
By phone number	Select the gateway port according to the called number (refer to the note below).
Cyclic Ascending	Select the next available channel in an ascending cycle order. Always select the next higher channel number in the Trunk Group. When the gateway reaches the highest channel number in the Trunk Group, it selects the lowest channel number in the Trunk Group and then starts ascending again (default).
Ascending	Select the lowest available channel. Always start at the lowest channel number in the Trunk Group and if that channel is not available, select the next higher channel.
Cyclic Descending	Select the next available channel in descending cycle order. Always select the next lower channel number in the Trunk Group. When the gateway reaches the lowest channel number in the Trunk Group, it selects the highest channel number in the Trunk Group and then start descending again.
Descending	Select the highest available channel. Always start at the highest channel number in the Trunk Group and if that channel is not available, select the next lower channel.
Number + Cyclic Ascending	First select the gateway port according to the called number (refer to the note below). If the called number isn't found, then select the next available channel in ascending cyclic order. Note that if the called number is found, but the port associated with this number is busy, the call is released.



**Note:** The internal numbers of the gateway's B-channels are defined in the 'Trunk Group Table' under the 'Phone Number' column. For detailed information on the 'Trunk Group Table', refer to Section 5.8.6 on page 58).

## 5.9 Advanced Configuration

Use this menu to set the gateway's advanced configuration parameters (for advanced users only).

### 5.9.1 Configuring the Network Settings

From the Network Settings page you can define:

- IP settings.
- NTP settings.
- Syslog settings.
- SNMP settings.
- RTP settings.
- Ethernet Ports Information (read-only).

➤ **To configure the Network Settings parameters, take these 4 steps:**

1. Open the 'Network Settings' screen (**Advanced Configuration** menu > **Network Settings**); the 'Network Settings' screen is displayed.
2. Configure the Network Settings parameters.
3. Click the **Submit** button to save your changes.
4. To save the changes so they are available after a power fail, refer to Section 5.12 on page 84.

**Figure 5-14: Network Settings Screen**

Network Settings	
<b>IP Settings</b>	
IP Address	10.8.58.4
Subnet Mask	255.255.0.0
Default Gateway Address	10.8.0.1
DNS Primary Server IP	0.0.0.0
DNS Secondary Server IP	0.0.0.0
Enable DHCP	Disable
NAT IP Address	0.0.0.0
<b>NTP Settings</b>	
NTP Server IP Address	0.0.0.0
NTP UTC Offset	Hours 0 Minutes 0
NTP Update Interval	Hours 24 Minutes 0
<b>Syslog Settings</b>	
Syslog Server IP Address	10.8.2.7
Enable Syslog	Enable

Note that the default **RTP Base UDP Port** is 6000.

### 5.9.1.1 Configuring the SNMP Managers Table

The SNMP Managers table allows you to configure the attributes of up to five SNMP managers.

➤ **To configure the SNMP Managers Table, take these 6 steps:**

1. Access the 'Network Settings' screen (**Advanced Configuration** menu > **Network Settings**); the 'Network Settings' screen is displayed (Figure 5-14).
2. Open the SNMP Managers Table screen by clicking the arrow sign (-->) to the right of the SNMP Managers Table label; the SNMP Managers Table screen is displayed (Figure 5-15).
3. Configure the SNMP managers parameters.
4. Click the **Submit** button to save your changes.
5. Click the **Close Window** button.
6. To save the changes so they are available after a power fail, refer to Section 5.12 on page 84.

**Figure 5-15: SNMP Managers Table Screen**

SNMP Managers Table*			
	IP Address	Trap Port	Trap Enable
<input type="checkbox"/> SNMP Manager 1	0.0.0.0	162	Enable
<input type="checkbox"/> SNMP Manager 2	0.0.0.0	162	Enable
<input type="checkbox"/> SNMP Manager 3	0.0.0.0	162	Enable
<input type="checkbox"/> SNMP Manager 4	0.0.0.0	162	Enable
<input type="checkbox"/> SNMP Manager 5	0.0.0.0	162	Enable



**Note:** If you clear a checkbox and click **Submit**, all settings in the same row revert to their defaults.

### 5.9.1.2 Multiple Routers Support

Multiple routers support is designed to assist the media gateway when it operates in a multiple routers network. The gateway learns the network topology by responding to ICMP redirections and caches them as routing rules (with expiration time).

When a set of routers operating within the same subnet serve as gateways to that network and intercommunicate using a dynamic routing protocol (such as OSPF, etc.), the routers can determine the shortest path to a certain destination and signal the remote host the existence of the better route. Using multiple router support the media gateway can utilize these router messages to change its next hop and establish the best path.

**Note:** Multiple Routers support is an integral feature that doesn't require configuration.

### 5.9.1.3 Simple Network Time Protocol Support

The Simple Network Time Protocol (SNTP) client functionality generates requests and reacts to the resulting responses using the NTP version 3 protocol definitions (according to RFC 1305). Through these requests and responses, the NTP client is able to synchronize the system time to a time source within the network, thereby eliminating any potential issues should the local system clock 'drift' during operation. By synchronizing time to a network time source, traffic handling,

maintenance, and debugging actions become simplified for the network administrator.

The NTP client follows a simple process in managing system time; the NTP client requests an NTP update, receives an NTP response, and updates the local system clock based on a configured NTP server within the network.

The client requests a time update from a specified NTP server at a specified update interval. In most situations this update interval should be every 24 hours based on when the system was restarted. The NTP server identity (as an IP address) and the update interval are configurable parameters that can be specified either in the *ini* file (NTPServerIP, NTPUpdateInterval respectively) or via an SNMP MIB object.

When the client receives a response to its request from the identified NTP server it must be interpreted based on time zone, or location, offset that the system is to a standard point of reference called the Universal Time Coordinate (UTC). The time offset that the NTP client should use is a configurable parameter that can be specified either in the *ini* file (NTPServerUTCOffset) or via an SNMP MIB object.

If required, the clock update is performed by the client as the final step of the update process. The update is done in such a way as to be transparent to the end users. For instance, the response of the server may indicate that the clock is running too fast on the client. The client slowly robs bits from the clock counter in order to update the clock to the correct time. If the clock is running too slow, then in an effort to catch the clock up, bits are added to the counter, causing the clock to update quicker and catch up to the correct time. The advantage of this method is that it does not introduce any disparity in the system time, that is noticeable to an end user, or that could corrupt call timeouts and timestamps.

## 5.9.2 Configuring the Channel Settings

The Channels Settings screen enables you to set the VoIP gateway channel parameters, such as Input and Output voice gain, Jitter buffer characteristics, Modem, Fax and DTMF transport modes. These parameters are applied to all Mediant 2000 channels.

Note that several Channels Settings parameters can be configured per call using profiles (refer to Section 5.8.5 on page 55).



**Note:** Channel parameters are changeable on-the-fly. Changes take effect from next call.

➤ **To configure the Channel Settings parameters, take these 4 steps:**

1. Open the 'Channel Settings' screen (**Advanced Configuration** menu > **Channel Settings**); the 'Channel Settings' screen is displayed.
2. Configure the Channel Settings parameters.
3. Click the **Submit** button to save your changes.
4. To save the changes so they are available after a power fail, refer to Section 5.12 on page 84.

**Figure 5-16: Channel Settings Screen**

Channel Settings	
<b>Voice Settings</b>	
Voice Volume (-32 to 31 dB)	1
Input Gain (-32 to 31 dB)	0
Silence Suppression	Disable
Echo Canceler	On
DTMF Transport Type	RFC2833 Relay DTMF
MF Transport Type	RFC2833 Relay MF
DTMF Volume (-31 to 0 dB)	-11
Enable Answer Detector	Disable
Answer Detector Activity Delay	0
Answer Detector Silence Time	10
Answer Detector Redirection	Disable
Answer Detector Sensitivity	0
<b>Fax/Modem/CID Settings</b>	
Fax Transport Mode	T.38 Relay
Caller ID Transport Type	Mute



**Note 1:** The parameters 'MF Transport Type' and the 5 Answer Detector parameters are not applicable to the Mediant 2000.

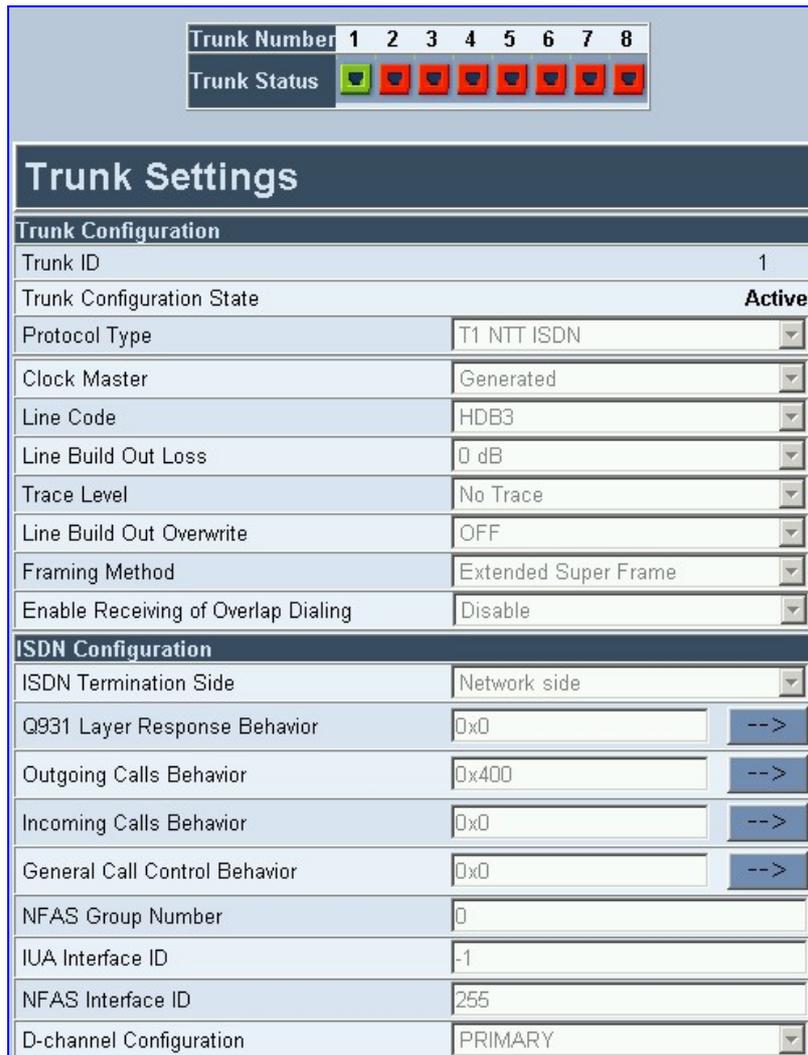
**Note 2:** The parameters 'DTMF Transport Type' and 'Fax Transport Mode' are overridden by the parameters 'IsDTMFUsed' and 'IsFaxUsed' respectively.

### 5.9.3 Configuring the Trunk Settings

➤ **To configure the Trunk Settings, take these 9 steps:**

1. Open the 'Trunk Settings' screen (**Advanced Configuration** menu > **Trunk Settings**); the 'Trunk Settings' screen is displayed.  
Initially, the screen appears with the parameters fields grayed (indicating read-only). The **Stop Trunk** button appears at the bottom of the screen.  
The Trunk Status indicators appear colored. [Table 5-7](#) shows the possible indicators and their descriptions.

**Figure 5-17: E1/T1 Trunk Settings Screen**



2. To configure the parameters of a specific trunk, from the trunks displayed on the top, select the trunk you want to configure by clicking the Trunk's Status indicator. The first parameter named 'Trunk ID' changes according to the trunk you click. The parameters displayed are for the selected trunk only.

Table 5-7: Trunks Status Color Indicator Keys

Indicator	Color	Description
	Gray	Disabled
	Green	Active-OK
	Yellow	RAI Alarm
	Red	LOS Alarm
	Blue	AIS Alarm
	Orange	D-channel Alarm (ISDN only)

- To modify the selected trunk's parameters, click the **Stop Trunk** button; the trunk is stopped, the status of the parameter 'Trunk Configuration State' changes to 'Non Active', the parameters are no longer grayed and can be modified and the **Apply Trunk Settings** button appears at the bottom of the screen.  
When all trunks are stopped, the **Apply to all Trunks** button also appears at the bottom of the screen.



**Note:** If the trunk can't be stopped because it provides the gateway's clock (assuming the Mediant 2000 is synchronized with the E1/T1 clock), assign a different E1/T1 trunk to provide the gateway's clock or enable 'TDM Bus PSTN Auto Clock' on the TDM Bus Settings screen.  
To assign a different E1/T1 trunk that provides the gateway's clock, access the 'TDM Bus Setting' screen and change the 'TDM Bus Local Reference' number to any other trunk number (this operation can be performed on-the-fly).

- Select the 'Protocol Type' you use. Note that different trunks can be defined with different protocols (CAS or ISDN variants) on the same gateway (subject to the constraints in the Mediant 2000 Release Notes).



**Note:** When modifying the 'Protocol Type' field, the menu is automatically updated according to the selected protocol (ISDN, CAS or Transparent). Additional parameters are appropriate to the selected protocol type.

- Modify the relevant trunk configuration parameters according to your requirements.
- To configure the different behavior bits: either enter the exact hexadecimal value of the bits in the field to the right of the relevant behavior parameter, or directly configure each bit field by completing the following steps:
  - Click the arrow button (-->) to the right of the relevant behavior parameter; a new window appears.
  - Modify each bit field according to your requirements.
  - Click the **Submit** button to save your changes.

- Click the **Close Window** button.
- 7. After modifying the parameters:
  - To apply the changes to the selected trunk only, click the **Apply Trunk Settings** button.
  - To apply the changes to all the trunks, click the **Apply to all Trunks** button.

The screen is refreshed, parameters become read-only (indicated by being grayed). The **Stop Trunk** button appears at the bottom of the screen.

- 8. To save the changes so they are available after a power fail, refer to Section 5.12 on page 84.



**Note:** Some parameter configuration options require a device reset; when this is the case, the Web Interface prompts the user.

- 9. To reset the Mediant 2000, refer to Section 5.12 on page 84.

### 5.9.4 Configuring the TDM Bus Settings

➤ **To configure the TDM Bus Settings parameters, take these 5 steps:**

1. Open the 'TDM Bus Settings' screen (**Advanced Configuration** menu > **TDM Bus Settings**); the 'TDM Bus Settings' screen is displayed.
2. Configure the TDM Bus Settings parameters.
3. Click the **Submit** button to save your changes.
4. To save the changes so they are available after a power fail, refer to Section 5.12 on page 84.
5. A device reset is required to activate the TDM Bus Settings parameters. To reset the Mediant 2000, refer to Section 5.12 on page 84.

**Figure 5-18: TDM Bus Settings Screen**

TDM Bus Settings	
Settings	
PCM Law Select	Alaw
TDM Bus Clock Source	Internal
TDM Bus Local Reference	1
TDM Bus PSTN Auto Clock	Disable
Idle PCM Pattern	85
Idle ABCD Pattern	15



**Note:** Usually the 'PCM Law Select' parameter is set to A-law for E1 trunks and to  $\mu$ -law for T1 trunks.

Refer to [Appendix E](#) on page 205 for information on configuring the 'TDM Bus Clock Source', 'TDM Bus Enable Fallback' and 'TDM Bus PSTN Auto Clock' parameters.

## 5.9.5 Restoring and Backing up the Gateway Configuration

The Configuration File screen enables you to restore (load a new *ini* file to the gateway) or to back up (make a copy of the VoIP gateway *ini* file and store it in a directory on your computer) the current configuration the gateway is using.

Back up your configuration if you want to protect your VoIP gateway programming. The backup *ini* file includes only those parameters that were modified and contain other than default values.

Restore your configuration if the VoIP gateway has been replaced or has lost its programming information, you can restore the VoIP gateway configuration from a previous backup or from a newly created *ini* file. To restore the VoIP gateway configuration from a previous backup you must have a backup of the VoIP gateway information stored on your computer.

### ➤ To restore or back up the *ini* file:

- Open the 'Configuration File' screen (**Advanced Configuration** menu > **Configuration File**); the 'Configuration File' screen is displayed.

Figure 5-19: Configuration File Screen

Configuration File

Get the *ini* file from the device to your computer

Get ini File

Send the *ini* file from your computer to the device

Browse...

Send ini File

The device will perform a 'Reset' after sending the *ini* file

### ➤ To back up the *ini* file, take these 4 steps:

1. Click the **Get ini File** button; the 'File Download' window opens.
2. Click the **Save** button; the 'Save As' window opens.
3. Navigate to the folder where you want to save the *ini* file.
4. Click the **Save** button; the VoIP gateway copies the *ini* file into the folder you selected.

### ➤ To restore the *ini* file, take these 4 steps:

1. Click the **Browse** button.
2. Navigate to the folder that contains the *ini* file you want to load.
3. Click the file and click the **Open** button; the name and path of the file appear in the field beside the Browse button.
4. Click the **Send ini File** button, and click **OK** in the prompt; the gateway is automatically reset (from the *cmp* version stored on the flash memory).

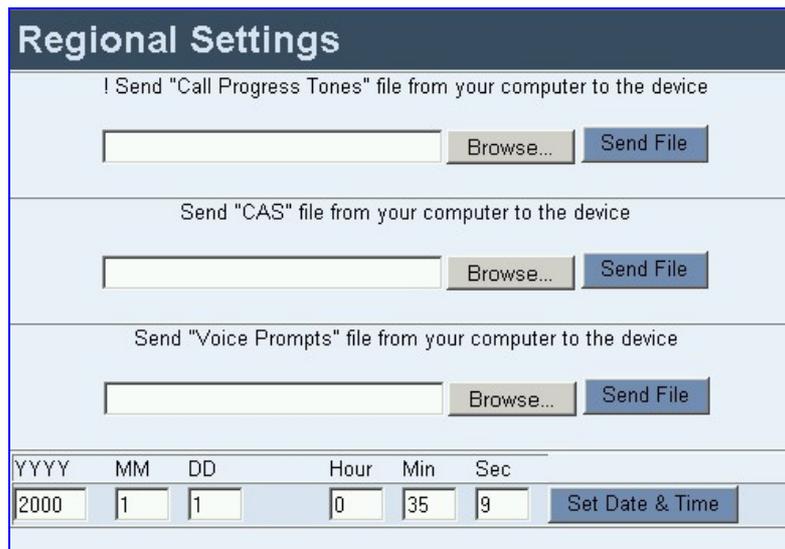
## 5.9.6 Regional Settings

The 'Regional Settings' screen enables you to set and view the gateway's internal date and time and to load to the gateway the following configuration files: Call Progress Tones, CAS and Voice Prompts. For detailed information on the configuration files, refer to Section 7 on page 135.

➤ **To configure the date and time of the Mediant 2000, take these 3 steps:**

1. Open the 'Regional Settings' screen (**Advanced Configuration** menu > **Regional Settings**); the 'Regional Settings' screen is displayed.

**Figure 5-20: Regional Settings Screen**



2. Enter the time and date where the gateway is installed.
3. Click the **Set Date & Time** button; the date and time are automatically updated.

Note that after performing a hardware reset, the date and time are returned to their defaults and should be updated.

➤ **To load a configuration file to the VoIP gateway, take these 8 steps:**

1. Open the 'Regional Settings' screen (**Advanced Configuration** menu > **Regional Settings**); the 'Regional Settings' screen is displayed (shown in Figure 5-20).
2. Click the **Browse** button adjacent to the file you want to load.
3. Navigate to the folder that contains the file you want to load.
4. Click the file and click the **Open** button; the name and path of the file appear in the field beside the **Browse** button.
5. Click the **Send File** button that is next to the field that contains the name of the file you want to load. An exclamation mark in the screen section indicates that the file's loading doesn't take effect on-the-fly (e.g., CPT file).
6. Repeat steps 2 to 5 for each file you want to load.



**Note 1:** Saving a configuration file to flash memory may disrupt traffic on the Mediant 2000. To avoid this, disable all traffic on the device before saving to flash memory.

**Note 2:** A device reset is required to activate a loaded CPT file.

7. To save the loaded auxiliary files so they are available after a power fail, refer to Section 5.12 on page 84.
8. To reset the Mediant 2000, refer to Section 5.12 on page 84.

## 5.9.7 Changing the Mediant 2000 Username and Password

To prevent unauthorized access to the Mediant 2000, it is recommended that you change the username and password (both are case-sensitive) that are used to access the Web Interface.

➤ **To change the username and password, take these 5 steps:**

1. Open the 'Change Password' screen (**Advanced Configuration** menu > **Change Password**); the 'Change Password' screen is displayed.

**Figure 5-21: Change Password Screen**



Change Password	
User Name	Admin
New Password	
Confirm Password	

2. In the 'User Name' and 'Password' fields, enter the new username and the new password respectively. Note that the username and password can be a maximum of 7 case-sensitive characters.
3. In the 'Confirm Password' field, reenter the new password.
4. Click the **Change Password** button; the new username and password are applied and the 'Enter Network Password' screen appears, shown in Figure 5-1 on page 41.
5. Enter the updated username and password in the 'Enter Network Password' screen.

## 5.10 Status & Diagnostic

Use this menu to view and monitor the gateway's channels, Syslog messages, hardware / software product information, and to assess the gateway's statistics and IP connectivity information.

### 5.10.1 Gateway Statistics

Use the screens under Gateway Statistics to monitor real-time activity such as IP Connectivity information, call details and call statistics, including the number of call attempts, failed calls, fax calls, etc.

**Note:** The Gateway Statistics screens doesn't refresh automatically. To view updated information re-access the screen you require.

#### 5.10.1.1 IP Connectivity

The IP Connectivity screen provides you with an online read-only network diagnostic connectivity information on all destination IP addresses configured in the Tel to IP Routing table.

**Note:** This information is available only if the parameter 'AltRoutingTel2IPEnable' (described in Table 6-5) is set to 1 (Enable) or 2 (Status Only).



**Note:** The information in columns 'Quality Status' and 'Quality Info.' (per IP address) is reset if two minutes elapse without a call to that destination.

- **To view the IP connectivity information, take these 2 steps:**
1. Set 'AltRoutingTel2IPEnable' to 1 or 2.
  2. Open the 'IP Connectivity' screen (**Status & Diagnostics** menu > **Gateway Statistics** submenu > **IP Connectivity**); the 'IP Connectivity' screen is displayed (Figure 5-22).

**Figure 5-22: IP Connectivity Screen**

IP Connectivity							
IP Address	Host Name	Connectivity Method	Connectivity Status	Quality Status	Quality Info.	DNS Status	
1	10.13.77.7	10.13.77.7	Ping	CON_OK	QOS_UNKNOWN	PL[percent]:0 DELAY [msec]:0	DNS_DISABLE
2	10.13.77.9	10.13.77.9	Ping	CON_OK	QOS_UNKNOWN	PL[percent]:0 DELAY [msec]:0	DNS_DISABLE
3	10.13.77.18	10.13.77.18	Ping	CON_FAIL	QOS_UNKNOWN	PL[percent]:0 DELAY [msec]:0	DNS_DISABLE
4	1.2.3.4	doron_pc	Ping	CON_FAIL	QOS_UNKNOWN	PL[percent]:0 DELAY [msec]:0	DNS_RESOLVED
5	10.13.2.95	xyz	Ping	CON_INIT	QOS_UNKNOWN	PL[percent]:0 DELAY [msec]:0	DNS_UNRESOLVED
6	UNUSED ENTRY	---	---	---	---	---	---
7	UNUSED ENTRY	---	---	---	---	---	---

**Table 5-8: IP Connectivity Parameters**

Column Name	Description
<b>IP Address</b>	IP address defined in the destination IP address field in the Tel to IP Routing table. or IP address that is resolved from the host name defined in the destination IP address field in the Tel to IP Routing table.
<b>Host Name</b>	Host name (or IP address) defined in the destination IP address field in the Tel to IP Routing table.
<b>Connectivity Method</b>	The method according to which the destination IP address is queried periodically (currently only by ping).
<b>Connectivity Status</b>	Displays the status of the IP address' connectivity according to the method in the 'Connectivity Method' field. Can be one of the following: <ul style="list-style-type: none"> <li>• OK = Remote side responds to periodic connectivity queries.</li> <li>• Lost = Remote side didn't respond for a short period.</li> <li>• Fail = Remote side doesn't respond.</li> <li>• Init = Connectivity queries not started (e.g., IP address not resolved).</li> <li>• Disable = The connectivity option is disabled ('AltRoutingTel2IPMode' equals 0 or 2).</li> </ul>
<b>Quality Status</b>	Determines the QoS (according to packet loss and delay) of the IP address. Can be one of the following: <ul style="list-style-type: none"> <li>• Unknown = Recent quality information isn't available.</li> <li>• OK</li> <li>• Poor</li> </ul> <p><b>Note 1:</b> This field is applicable only if the parameter 'AltRoutingTel2IPMode' is set to 2 or 3. <b>Note 2:</b> This field is reset if no QoS information is received for 2 minutes.</p>

**Table 5-8: IP Connectivity Parameters**

Column Name	Description
Quality Info.	Displays QoS information: delay and packet loss, calculated according to previous calls. <b>Note 1:</b> This field is applicable only if the parameter 'AltRoutingTel2IPMode' is set to 2 or 3. <b>Note 2:</b> This field is reset if no QoS information is received for 2 minutes.
DNS Status	Can be one of the following: <ul style="list-style-type: none"> <li>• DNS Disable</li> <li>• DNS Resolved</li> <li>• DNS Unresolved</li> </ul>

**5.10.1.2 Call Counters**

The Call Counters screens provide you with statistic information on incoming (IP→Tel) and outgoing (Tel→IP) calls. The statistic information is updated according to the release reason that is received after a call is terminated (during the same time as the end-of-call CDR message is sent). The release reason can be viewed in the Termination Reason field in the CDR message. For detailed information on each counter, refer to [Table 5-9](#) on page 73.

You can reset this information by clicking the **Reset Counters** button.

➤ **To view the IP→Tel and Tel→IP Call Counters information:**

- Open the Call Counters screen you want to view (**Status & Diagnostics** menu > **Gateway Statistics** submenu); the relevant Call Counters screen is displayed. [Figure 5-23](#) shows the 'Tel→IP Call Counters' screen.

**Figure 5-23: Tel→IP Call Counters Screen**

Tel to IP Calls Count	
Number of Attempted Calls	10
Number of Established Calls	5
Percentage of Successful Calls	50.000000
Number of Failed Calls due to a Busy Line	1
Number of Failed Calls due to No Answer	3
Number of Failed Calls due to No Route	0
Number of Failed Calls due to No Matched Capabilities	0
Number of Failed Calls due to Other Failures	1
Average Call Duration [sec]	15
Attempted Fax Calls Counter	0
Successful Fax Calls Counter	0

**Table 5-9: Call Counters Description (continues on pages 73 to 74)**

Counter	Description
Number of Attempted Calls	This counter indicates the number of attempted calls. It is composed of established and failed calls. The number of established calls is represented by the 'Number of Established Calls' counter. The number of failed calls is represented by the five failed-call counters. Only one of the established / failed call counters is incremented every time.

**Table 5-9: Call Counters Description (continues on pages 73 to 74)**

Counter	Description
Number of Established Calls	This counter indicates the number of established calls. It is incremented as a result of one of the following release reasons, if the duration of the call is bigger then zero: GWAPP_REASON_NOT_RELEVANT (0) GWAPP_NORMAL_CALL_CLEAR (16) GWAPP_NORMAL_UNSPECIFIED (31) And the internal reasons: RELEASE_BECAUSE_UNKNOWN_REASON RELEASE_BECAUSE_REMOTE_CANCEL_CALL RELEASE_BECAUSE_MANUAL_DISC RELEASE_BECAUSE_SILENCE_DISC RELEASE_BECAUSE_DISCONNECT_CODE <b>Note:</b> When the duration of the call is zero, the release reason GWAPP_NORMAL_CALL_CLEAR increments the 'Number of Failed Calls due to No Answer' counter. The rest of the release reasons increment the 'Number of Failed Calls due to Other Failures' counter.
Number of Failed Calls due to a Busy Line	This counter indicates the number of calls that failed as a result of a busy line. It is incremented as a result of the following release reason: GWAPP_USER_BUSY (17)
Number of Failed Calls due to No Answer	This counter indicates the number of calls that weren't answered. It is incremented as a result of one of the following release reasons: GWAPP_NO_USER_RESPONDING (18) GWAPP_NO_ANSWER_FROM_USER_ALERTED (19)  And (when the call duration is zero) as a result of the following: GWAPP_NORMAL_CALL_CLEAR (16) RELEASE_BECAUSE_NORMAL_CALL_DROP (internal)
Number of Failed Calls due to No Route	This counter indicates the number of calls whose destinations weren't found. It is incremented as a result of one of the following release reasons: GWAPP_UNASSIGNED_NUMBER (1) GWAPP_NO_ROUTE_TO_DESTINATION (3)
Number of Failed Calls due to No Matched Capabilities	This counter indicates the number of calls that failed due to mismatched gateway capabilities. It is incremented as a result of an internal identification of capability mismatch. This mismatch is reflected to CDR via the value of the parameter 'DefaultReleaseReason' (default is GWAPP_NO_ROUTE_TO_DESTINATION (3)), or by the GWAPP_SERVICE_NOT_IMPLEMENTED_UNSPECIFIED(79) reason.
Number of Failed Calls due to Other Failures	This counter is incremented as a result of calls that fail due to reasons not covered by the other counters.
Percentage of Successful Calls	The percentage of established calls from attempted calls.
Average Call Duration [sec]	The average call duration of established calls.
Attempted Fax Calls Counter	This counter indicates the number of attempted fax calls.
Successful Fax Calls Counter	This counter indicates the number of successful fax calls.

### 5.10.2 Monitoring the Mediant 2000 Trunks & Channels

The Trunk & Channel Status screen provides real time monitoring on the current status of the Mediant 2000 trunks & channels.

- **To monitor the status of the trunks and B-channels take this step:**
  - Open the 'Trunk & Channel Status' screen (**Status & Diagnostics** menu > **Channel Status**); the 'Trunk & Channel Status' screen is displayed.

**Figure 5-24: Mediant 2000 Trunk & Channel Status Screen**

Trunks	Channels
Trunk Status	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
 Trunk 1	                        
 Trunk 2	                        
 Trunk 3	                        
 Trunk 4	                        
 Trunk 5	                        
 Trunk 6	                        
 Trunk 7	                        
 Trunk 8	                        

The number of trunks and channels that appear on the screen depends on the system configuration. The example above depicts a system with 8 T1 spans.

The trunk and channel status indicators appear colored. [Figure 5-25](#) shows the possible indicators and their descriptions.

**Figure 5-25: Trunk and Channel Status Color Indicator Keys**

Trunk	Channel
 Disable	 Inactive
 Active - OK	 Active
 RAI Alarm	 SS7
 LOS Alarm	 Non Voice
 AIS Alarm	
 D-channel Alarm	

- **To monitor the details of a specific channel, take these 2 steps:**
  1. Click the numbered icon of the specific channel whose detailed status you need to check/monitor; the channel-specific Channel Status screen appears, shown in [Figure 5-26](#).

2. Click the submenu links to check/view a specific channel's parameter settings.

Figure 5-26: Channel Status Details Screen

SIP Channel Status		
<b>Static Information</b>		
Endpoint Status :	ACTIVE	
Assigned Phone Number :	100	
Trunk Group :	default (0)	
MWI Information :	--	
<b>Associated Calls Information</b>		
Call ID :	265821508dMlu@10.8.58.1	--
Call Originator :	TEL	--
Source Tel Number :	100	--
Destination Tel Number :	200	--
Redirect Calling Number :		--
Remote Signaling IP :	10.8.58.2	--
Remote RTP (IP:Port) :	10.8.58.2: 4000	--
Call Establishment Duration :	2	--
Call Duration :	17	--
Call State :	SESSION	--
Fax State :	n/a	--
Coder + PTime :	g7231:30	--
Call Type :	Voice	--
Call Establishment Method :	Normal	--
DTMF Selected Method for Tx/Rx :	DTMF_NOT_SUPPORTED	--

### 5.10.3 Activating the Internal Syslog Viewer

The Message Log screen displays Syslog debug messages sent by the gateway.

Note that it is not recommended to keep a 'Message Log' session open for a prolonged period (refer to the Note below). For prolong debugging use an external Syslog server, refer to Section 9.2 on page 165.

Refer to the Debug Level parameter 'GwDebugLevel' (described in Table 6-1) to determine the Syslog logging level.

➤ **To activate the Message Log, take these 4 steps:**

1. In the **General Parameters** screen under **Advanced Parameters** submenu (accessed from the **Protocol Management** menu), set the parameter 'Debug Level' to 5. This parameter determines the Syslog logging level, in the range 0 to 5, where 5 is the highest level.
2. Open the 'Message Log' screen (**Status & Diagnostics** menu > **Message Log**); the 'Message Log' screen is displayed.

Figure 5-27: Message Log Screen

```

Log is Activated

12d:6h:56m:26s ( lgr_flow)(460 ) ---- Incoming SIP Message from 10.8.58.1:5060 ----

12d:6h:56m:26s INVITE sip:200@10.8.58.4;user=phone SIP/2.0
Via: SIP/2.0/UDP 10.8.58.1;branch=z9hG4bKackpUGBoT
Max-Forwards: 70
From: <sip:100@10.8.58.1>;tag=1c910315947
To: <sip:200@10.8.58.4;user=phone>
Call-ID: 1254421147LEqU@10.8.58.1
CSeq: 1 INVITE
Contact: <sip:100@10.8.58.1>
Supported: em,timer,replaces,path
Allow: REGISTER,OPTIONS,INVITE,ACK,CANCEL,BYE,NOTIFY,PRACK,REFER,INFO,SUBSCRIBE,UPDATE
User-Agent: Audiocodes-Sip-Gateway-MP-104 FXS/v.4.40.123.223
Content-Type: application/sdp
Content-Length: 161
    
```

3. Select the messages, copy them and paste them into a text editor such as Notepad. Send this *txt* file to our Technical Support for diagnosis and troubleshooting.
4. To clear the screen of messages, click on the submenu **Message Log**; the screen is cleared and new messages begin appearing.



**Tip:** Do not keep the 'Message Log' screen minimized for a prolonged period as a prolonged session may cause the Mediant 2000 to overload. As long as the screen is open (even if minimized), a session is in progress and messages are sent. Closing the screen (and accessing another) stops the messages and terminates the session.

### 5.10.4 System Information

The System Information screen displays specific hardware and software product information. This information can help you to expedite any troubleshooting process. Capture the screen and email it to 'our' Technical Support personnel to ensure quick diagnosis and effective corrective action. From this screen you can also view and remove any loaded auxiliary files used by the Mediant 2000 (stored in the RAM).

➤ **To access the System Information screen:**

- Open the 'System Information' screen (**Status & Diagnostics** menu > **System Information**); the 'System Information' screen is displayed.

**Figure 5-28: System Information Screen**

System Information	
<b>General</b>	
MAC Address:	00908f036a4c
Serial Number:	223820
Board Type:	24
System Up Time:	0d:15h:49m:5s:60th
<b>Versions</b>	
Version ID:	4.40.123.218
DSP Type:	2
DSP Software Version:	20606
DSP Software Name:	624AE3
Flash Version:	188
Module FirmWare:	0x10
<b>Loaded Files</b>	
Call Progress Tone File Name:	usa_tones_09.dat <input type="button" value="Delete"/>

➤ **To delete any of the loaded auxiliary files, take these 3 steps:**

1. Press the **Delete** button to the right of the files you want to delete. Deleting a file takes effect only after the Mediant 2000 is reset.
2. Click the **Reset** button on the main menu bar; the Reset screen is displayed.
3. Select the **Burn** option and click the **Reset** button. The Mediant 2000 is reset and the auxiliary files you chose to delete are discarded.

## 5.11 Software Update Menu

The 'Software Update' menu enables users to upgrade the Mediant 2000 software by loading a new *cmp* file along with the *ini* and a suite of auxiliary files, or to update the existing auxiliary files.

The 'Software Update' menu comprises two submenus:

- Software Update Wizard (refer to Section 5.11.1 below).
- Auxiliary Files (refer to Section 5.11.2 on page 82).



**Note:** When upgrading the Mediant 2000 software you *must* load the new *cmp* file with all other related configuration files

### 5.11.1 Software Upgrade Wizard

The Software Upgrade Wizard guides users through the process of software upgrade: selecting files and loading them to the gateway. The wizard also enables users to upgrade software while maintaining the existing configuration. Using the wizard obligates users to load a *cmp* file. Users can choose to also use the Wizard to load the *ini* and auxiliary files (e.g., Call Progress Tones) but this option cannot be pursued without loading the *cmp* file. For the *ini* and each auxiliary file type, users can choose to reload an existing file, load a new file or not load a file at all.



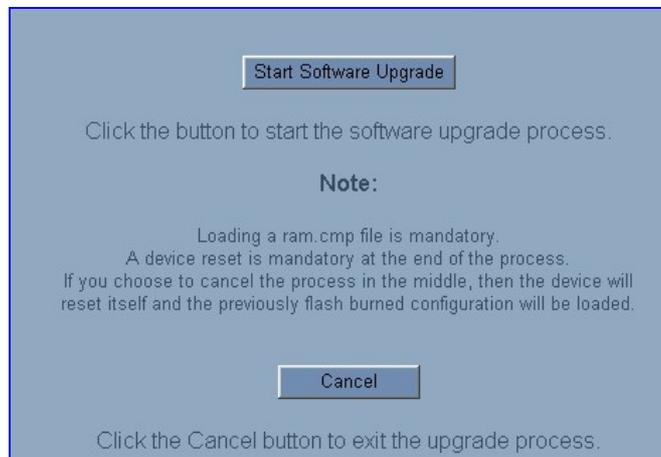
**Warning 1:** The Software Upgrade Wizard requires the Mediant 2000 to be reset at the end of the process, disrupting any of its traffic. To avoid disruption, disable all traffic on the Mediant 2000 before initiating the Wizard.

**Warning 2:** Verify, prior to clicking the Start Software Upgrade button that no traffic is running on the device. After clicking this button a device reset is mandatory. Even if you choose to cancel the process in the middle, the device resets itself and the previous configuration burned to flash is reloaded.

➤ **To use the Software Upgrade Wizard, take these 9 steps:**

1. Stop all traffic on the Mediant 2000 (refer to the note above).
2. Open the 'Software Upgrade Wizard' (**Software Update** menu > **Software Upgrade Wizard**); the 'Start Software Upgrade' screen appears.

**Figure 5-29: Start Software Upgrade Screen**





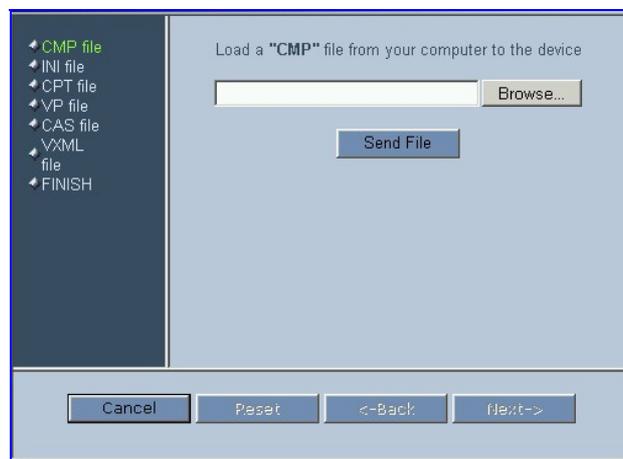
**Note:** At this point, the process can be canceled with no consequence to the Mediant 2000 (click the **Cancel** button). If you continue the process (by clicking the **Start Software Upgrade** button), the process must be followed through and completed with a Mediant 2000 reset at the end. If you click the **Cancel** button in any of the subsequent screens, the Mediant 2000 is automatically reset with the configuration that was previously burned in flash memory.

- Click the **Start Software Upgrade** button; the 'Load a *cmp* file' screen appears (Figure 5-30).



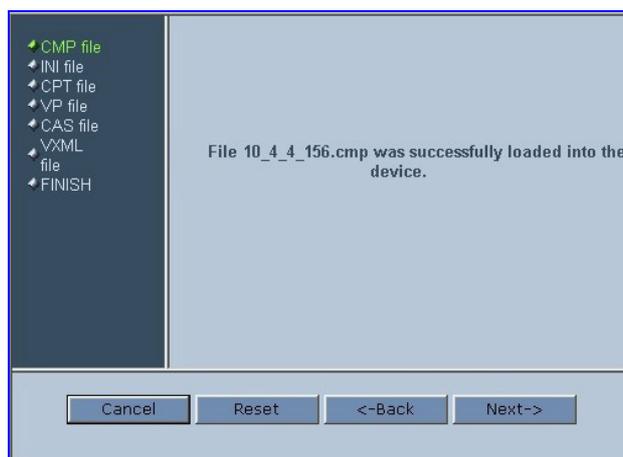
**Note:** When in the Wizard process, the rest of the Web application is unavailable and the background Web screen is disabled. After the process is completed, access to the full Web application is restored.

**Figure 5-30: Load a *cmp* File Screen**



- Click the **Browse** button, navigate to the *cmp* file and click the button **Send File**; the *cmp* file is loaded to the Mediant 2000 and you're notified as to a successful loading (refer to Figure 5-31).

**Figure 5-31: *cmp* File Successfully Loaded into the Mediant 2000 Notification**

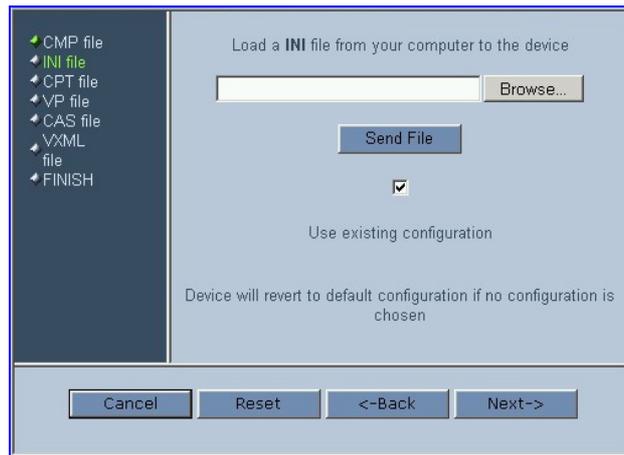


- Note that the four action buttons (**Cancel**, **Reset**, **Back**, and **Next**) are now activated (following *cmp* file loading). You can now choose to either:
  - Click **Reset**; the Mediant 2000 resets, utilizing the new *cmp* you loaded and utilizing the current configuration files.

- Click **Cancel**; the Mediant 2000 resets utilizing the *cmp*, *ini* and all other configuration files that were previously stored in flash memory. Note that these are NOT the files you loaded in the previous Wizard steps.
- Click **Back**; the 'Load a *cmp* File' screen is reverted to; refer to [Figure 5-30](#).
- Click **Next**; the 'Load an *ini* File' screen opens; refer to [Figure 5-32](#). Loading a new *ini* file or any other auxiliary file listed in the Wizard is optional.

Note that as you progress, the file type list on the left indicates which file type loading is in process by illuminating green (until 'FINISH').

**Figure 5-32: Load an *ini* File Screen**

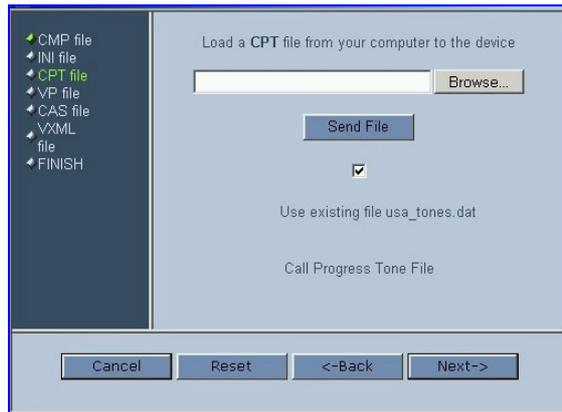


6. In the 'Load an *ini* File' screen, you can now choose to either:
  - Click **Browse** and navigate to the *ini* file; the check box 'Use existing configuration', by default checked, becomes unchecked. Click **Send File**; the *ini* file is loaded to the Mediant 2000 and you're notified as to a successful loading.
  - Ignore the **Browse** button (its field remains undefined and the check box 'Use existing configuration' remains checked by default).
  - Ignore the **Browse** button and uncheck the 'Use existing configuration' check box; no *ini* file is loaded, the Mediant 2000 uses its factory-preconfigured values.

You can now choose to either:

- Click **Cancel**; the Mediant 2000 resets utilizing the *cmp*, *ini* and all other configuration files that were previously stored in flash memory. Note that these are NOT the files you loaded in the previous Wizard steps.
- Click **Reset**; the Mediant 2000 resets, utilizing the new *cmp* and *ini* file you loaded up to now as well as utilizing the other configuration files.
- Click **Back**; the 'Load a *cmp* file' screen is reverted to; refer to [Figure 5-30](#).
- Click **Next**; the 'Load a CPT File' screen opens, refer to [Figure 5-33](#); Loading a new CPT file or any other auxiliary file listed in the Wizard is optional.

**Figure 5-33: Load a CPT File Screen**



7. Follow the same procedure you followed when loading the *ini* file (refer to Step 6). The same procedure applies to the 'Load a VP file' (not applicable to the Mediant 2000 gateway) screen and 'Load a coefficient file' screen.
8. In the 'FINISH' screen (refer to Figure 5-34), the **Next** button is disabled. Complete the upgrade process by clicking **Reset** or **Cancel**.

Button	Result
<b>Reset</b>	The Mediant 2000 'burns' the newly loaded files to flash memory. The 'Burning files to flash memory' screen appears. Wait for the 'burn' to finish. When it finishes, the 'End Process' screen appears displaying the burned configuration files (refer to Figure 5-35).
<b>Cancel</b>	The Mediant 2000 resets, utilizing the files previously stored in flash memory. (Note that these are NOT the files you loaded in the previous Wizard steps).

**Figure 5-34: FINISH Screen**

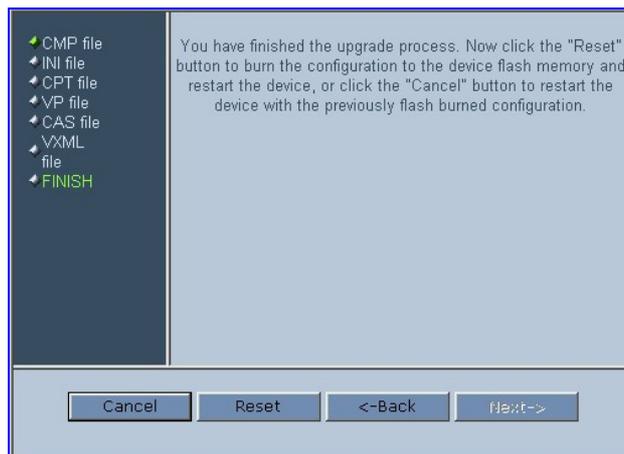
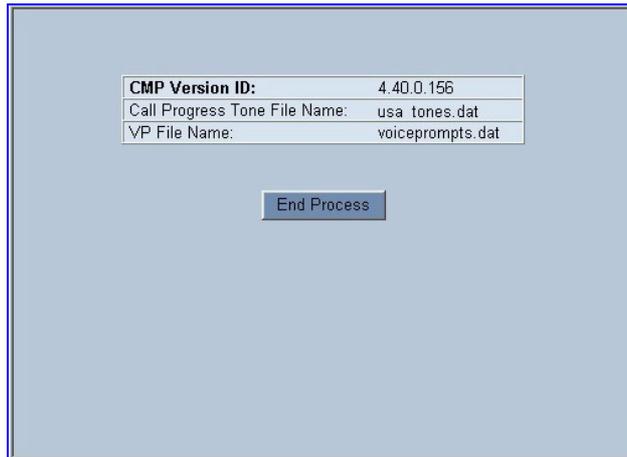


Figure 5-35: 'End Process' Screen



9. Click the **End Process** button; the 'Quick Setup' screen appears and the full Web application is reactivated.

### 5.11.2 Auxiliary Files

The 'Auxiliary Files' screen enables you to load to the gateway the following files: CAS, Call Progress Tones, Voice Prompts and Prerecorded Tones (PRT). For detailed information on these files, refer to Section 7 on page 135. For information on deleting these files from the Mediant 2000, refer to Section 5.10.4 on page 77.

Table 5-10 presents a brief description of each auxiliary file.

Table 5-10: Auxiliary Files Descriptions

File Type	Description
<b>CAS</b>	Up to 8 different CAS files containing specific CAS protocol definitions. These files are provided to support various types of CAS signaling.
<b>Voice Prompts</b>	The voice announcement file contains a set of Voice Prompts to be played by the Mediant 2000 during operation. Applicable only to the VXML application.
<b>Call Progress Tones</b>	This is a region-specific, telephone exchange-dependent file that contains the Call Progress Tones levels and frequencies that the VoIP gateway uses. The default CPT file is: U.S.A.
<b>Prerecorded Tones</b>	The <i>dat</i> PRT file enhances the gateway's capabilities of playing a wide range of telephone exchange tones that cannot be defined in the Call Progress Tones file.

➤ **To load an auxiliary file to the gateway, take these 8 steps:**

1. Open the 'Auxiliary Files' screen (**Software Upgrade** menu > **Load Auxiliary Files**); the 'Auxiliary Files' screen is displayed.

Figure 5-36: Auxiliary Files Screen

2. Click the **Browse** button that is in the field for the type of file you want to load.
3. Navigate to the folder that contains the file you want to load.
4. Click the file and click the **Open** button; the name and path of the file appear in the field beside the **Browse** button.
5. Click the **Send File** button that is next to the field that contains the name of the file you want to load. An exclamation mark in the screen section indicates that the file's loading doesn't take effect on-the-fly (e.g., CPT file).
6. Repeat steps 2 to 5 for each file you want to load.



**Note 1:** Saving an auxiliary file to flash memory may disrupt traffic on the Mediant 2000. To avoid this, disable all traffic on the device before saving to flash memory.

**Note 2:** A device reset is required to activate a loaded CPT file, and may be required for the activation of certain *ini* file parameters.

7. To save the loaded auxiliary files so they are available after a power fail, refer to Section 5.12 on page 84.
8. To reset the Mediant 2000, refer to Section 5.12 on page 83.

### 5.11.3 Updating the Software Upgrade Key

Mediant 2000 gateways are supplied with a Software Upgrade Key already pre-configured for each of its TrunkPack Modules (TPM).

Users can later upgrade their Mediant 2000 features, capabilities and quantity of available resources by specifying what upgrades they require, and purchasing a new key to match their specification.

The Software Upgrade Key is sent as a string in a text file, to be loaded into the Mediant 2000. Stored in the device's non-volatile flash memory, the string defines the features and capabilities allowed by the specific key purchased by the user. The device allows users to utilize *only these* features and capabilities. A new key overwrites a previously installed key.

For detailed information on the Software Upgrade Key, refer to [Appendix H](#) on page 223.

## 5.12 Save Configuration

The Save Configuration screen enables users to save the current parameter configuration and the loaded auxiliary files to the *non-volatile* memory so they are available after a power fail. Parameters that are only saved to the *volatile* memory revert to their previous settings after hardware reset.

Note that when performing a software reset (i.e., via Web or SNMP) you can choose to save the changes to the *non-volatile* memory. Therefore, there is no need to use the Save Configuration screen.



**Note:** Saving changes to the *non-volatile* memory may disrupt traffic on the gateway. To avoid this, disable all traffic before saving.

➤ **To save the changes to the *non-volatile*, take these 2 steps:**

1. Click the **Save Configuration** button on the main menu bar; the 'Save Configuration' screen is displayed.

**Figure 5-37: Save Configuration Screen**



2. Click the **Save Configuration** button in the middle of the screen; a confirmation message appears when the save is complete.

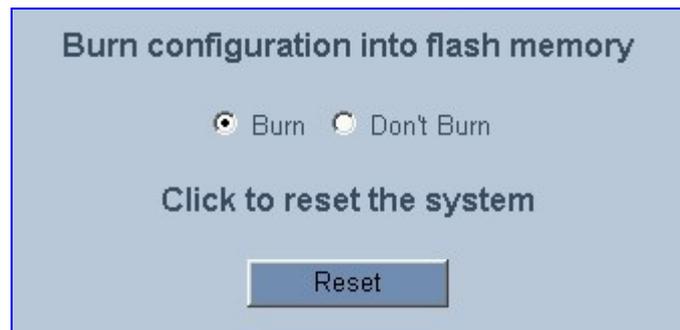
## 5.13 Resetting the Mediant 2000

The Reset screen enables you to remotely reset the gateway. Before reset you can choose to save the gateway configuration to flash memory.

➤ **To reset the Mediant 2000, take these 3 steps:**

1. Click the **Reset** button on the main menu bar; the Reset screen is displayed.

**Figure 5-38: Reset Screen**



2. Select one of the following options:
  - Burn - (default) the current configuration is burned to flash prior to reset.
  - Don't Burn - resets the device without burning the current configuration to flash (discards all modifications to the configuration).
3. Click the **Reset** button. If the Burn option is selected, all configuration changes are saved to flash memory. If the Don't Burn option is selected, all configuration changes are discarded. The device is shut down and re-activated. A message about the waiting period is displayed. The screen is refreshed.



**Note:** When Gatekeeper is used, the gateway issues an Unregister request before it is reset (either from the Embedded Web Server, SNMP or BootP).

## Reader's Notes

## 6 *ini* File Configuration of the Mediant 2000

As an alternative to configuring the VoIP gateway using the Web Interface (refer to Section 5 on page 39), it can be configured by loading the *ini* file containing Customer-configured parameters.

The *ini* file is loaded via the BootP/TFTP utility (refer to Appendix B on page 189) or via any standard TFTP server. It can also be loaded through the Web Interface (refer to Section 5.9.5 on page 69).

The *ini* file configuration parameters are stored in the Mediant 2000 non-volatile memory after the file is loaded. When a parameter is missing from the *ini* file, a default value is assigned to that parameter (according to the *cmp* file loaded on the Mediant 2000) and stored in the non-volatile memory (thereby overriding the value previously defined for that parameter). Therefore, to restore the default configuration parameters, use the *ini* file without any valid parameters or with a semicolon (;) preceding all lines in the file.

Some of the Mediant 2000 parameters are configurable through the *ini* file only (and not via the Web). These parameters usually determine a low-level functionality and are seldom changed for a specific application.

### 6.1 Secured *ini* File

The *ini* file contains sensitive information that is required for the functioning of the Mediant 2000. It is loaded to, or retrieved from, the device via TFTP or HTTP. These protocols are unsecured and vulnerable to potential hackers. Therefore an encoded *ini* file significantly reduces these threats.

You can choose to load an encoded *ini* file to the Mediant 2000. When you load an encoded *ini* file, the retrieved *ini* file is also encoded. Use the 'TrunkPack Downloadable Conversion Utility' to encode or decode the *ini* file before you load it to, or retrieve it from, the device. Note that the encoded *ini* file's loading procedure is identical to the regular *ini* file's loading procedure. For information on encoding / decoding an *ini* file, refer to Section G.1.3 on page 217.

### 6.2 Modifying an *ini* File

➤ **To modify the *ini* file, take these 3 steps:**

1. Get the *ini* file from the gateway using the Embedded Web Server (refer to Section 5.9.5 on page 69).
2. Open the file (the file opens in Notepad or a Customer-defined text file editor) and modify the *ini* file parameters according to your requirements. Save and close the file.
3. Load the modified *ini* file to the gateway (using either the BootP/TFTP utility or the Embedded Web Server).

This method preserves the programming that already exists in the device, including special default values that were preconfigured when the unit was manufactured.



**Tip:** Before loading the *ini* file to the gateway, verify that the extension of the *ini* file saved on your PC is correct: Verify that the check box 'Hide file extension for known file types' (My computer>Tools>Folder Options>View) is unchecked. Then, confirm that the *ini* file name extension is xxx.ini and NOT erroneously xxx.ini.ini or xxx~.ini.

## 6.3 The *ini* File Content

The *ini* file contains the following SIP gateway information:

- Basic, Logging, Web and RADIUS parameters shown in [Table 6-1](#) on page 90.
- SNMP parameters shown in [Table 6-2](#) on page 98.
- SIP Configuration parameters shown in [Table 6-3](#) on page 100.
- ISDN and CAS Interworking-Related Parameters shown in [Table 6-4](#) on page 111.
- Number Manipulation and Routing parameters shown in [Table 6-5](#) on page 115.
- E1/T1 Configuration Parameters shown in [Table 6-6](#) on page 122.
- Channel Parameters shown in [Table 6-7](#) on page 128.
- Configuration Files parameters shown in [Section 6.12.1](#) on page 132.

## 6.4 The *ini* File Structure

The *ini* file can contain any number of parameters. The parameters are divided into groups by their functionality. The general form of the *ini* file is shown in [Figure 6-1](#) below.

**Figure 6-1: *ini* File Structure**

```
[Sub Section Name]

Parameter_Name = Parameter_Value
Parameter_Name = Parameter_Value

; REMARK

[Sub Section Name]
```

### 6.4.1 The *ini* File Structure Rules

- Lines beginning with a semi-colon ';' (as the first character) are ignored.
- A Carriage Return must be the final character of each line.
- The number of spaces before and after "=" is not relevant.
- If there is a syntax error in the parameter name, the value is ignored.
- Syntax errors in the parameter value field can cause unexpected errors (because parameters may be set to the wrong values).
- Sub-section names are optional.
- String parameters, representing file names, for example CallProgressTonesFileName, must be placed between two inverted commas ('...').
- The parameter name is NOT case-sensitive; the parameter value is NOT case-sensitive *except for coder names*.
- The *ini* file should be ended with one or more carriage returns.

## 6.5 The *ini* File Example

Figure 6-2 shows an example of an *ini* file for the VoIP gateway.

**Figure 6-2: SIP *ini* File Example**

```
PCMLawSelect = 1
ProtocolType = 1
TerminationSide = 0
FramingMethod = 0
LineCode = 2
TDMBusClockSource = 4
ClockMaster = 0

;Channel Params
DJBufferMinDelay = 75
RTPRedundancyDepth = 1

IsProxyUsed = 1
ProxyIP = 192.168.122.179

CoderName = g7231,90

;List of serial B-channel numbers

TrunkGroup_1 = 0/1-24,1000
TrunkGroup_2 = 1/1-24,2000
TrunkGroup_3 = 2/1-24,3000
TrunkGroup_4 = 3/1-24,4000

EnableSyslog = 1
SyslogServerIP = 10.2.2.1

CallProgressTonesFilename = 'CPUSA.dat'
;CASFileName = 'E_M_WinkTable.dat'
SaveConfiguration = 1
```

## 6.6 Basic, Logging, Web and RADIUS Parameters



**Note:** In Table 6-1, parameters in brackets are the format in the Embedded Web Server .

Table 6-1: Basic, Logging, Web and RADIUS Parameters (continues on pages 90 to 97)

<i>ini</i> File Field Name Web Parameter Name *	Valid Range and Description
<b>EthernetPhyConfiguration</b>	0 = 10 Base-T half-duplex 1 = 10 Base-T full-duplex 2 = 100 Base-TX half-duplex 3 = 100 Base-TX full-duplex 4 = auto-negotiate (Default) Auto-negotiate falls back to half-duplex mode (HD) when the opposite port is not in auto-negotiate, but the speed (10 Base-T, 100 Base -TX) in this mode is always configured correctly.
<b>DHCPEnable</b> [Enable DHCP]	0 = Disable DHCP support on the gateway (default). 1 = Enable DHCP support on the gateway.  After the gateway is powered up, it attempts to communicate with a BootP server. If a BootP server is not responding and if DHCP is enabled, then the gateway attempts to get its IP address and other network parameters from the DHCP server.  <b>Web Note:</b> After you enable the DHCP Server (from the Web browser) follow this procedure: <ul style="list-style-type: none"> <li>Click the Submit button.</li> <li>Save the configuration using the 'Save Configuration' button (before you reset the gateway). For information on how to save the configuration, refer to Section 5.12 on page 84.</li> <li>Reset the gateway <i>directly</i> (Web reset doesn't trigger the BootP/DHCP procedure and the parameter DHCPEnable reverts to '0').</li> </ul> Note that throughout the DHCP procedure the BootP/TFTP application must be deactivated. Otherwise, the gateway receives a response from the BootP server instead of the DHCP server.  <b>Note:</b> The DHCPEnable is a special "Hidden" parameter. Once defined and saved in flash memory, its assigned value doesn't revert to its default even if the parameter doesn't appear in the <i>ini</i> file.
<b>EnableDiagnostics</b>	0 = No diagnostics (default) 1 = Perform diagnostics
<b>EnableLanWatchDog</b> [Enable LAN Watchdog]	0 = Disable LAN Watch-Dog (default). 1 = Enable LAN Watch-Dog. If LAN Watch-Dog is enabled, the gateway restarts when a network failure is detected.
<b>DNSPriServerIP</b> [DNS Primary Server IP]	IP address of the primary DNS server in dotted format notation.
<b>DNSSecServerIP</b> [DNS Secondary Server IP]	IP address of the secondary DNS server in dotted format notation.

Table 6-1: Basic, Logging, Web and RADIUS Parameters (continues on pages 90 to 97)

<i>ini</i> File Field Name Web Parameter Name	Valid Range and Description
<b>DNS2IP</b> [Internal DNS Table]	<p>Internal DNS table, used to resolve host names to IP addresses. Two different IP addresses (in dotted format notation) can be assigned to a hostname.</p> <p>DNS2IP = &lt;Hostname&gt;, &lt;first IP address&gt;, &lt;second IP address&gt;</p> <p><b>Note 1:</b> If the internal DNS table is configured, the gateway first tries to resolve a domain name using this table. If the domain name isn't found, the gateway performs a DNS resolution using an external DNS server.</p> <p><b>Note 2:</b> This parameter can appear up to 10 times.</p>
<b>GWAppDelayTime</b> [Delay After Reset [sec]]	<p>Defines the amount of time (in seconds) the gateway's operation is delayed after a reset cycle.</p> <p>The valid range is 0 to 600. The default value is 5 seconds.</p> <p><b>Note:</b> This feature helps to overcome connection problems caused by some LAN routers or IP configuration parameters change by a DHCP Server.</p>
<b>DisableNAT</b>	<p>Enables / disables the Network Address Translation (NAT) mechanism.</p> <p>0 = Enabled.</p> <p>1 = Disabled (default).</p> <p><b>Note:</b> The compare operation that is performed on the IP address is enabled by default and is controlled by the parameter 'EnableIPAddrTranslation'. The compare operation that is performed on the UDP port is disabled by default and is controlled by the parameter 'EnableUDPPortTranslation'.</p>
<b>EnableIPAddrTranslation</b>	<p>0 = Disable IP address translation.</p> <p>1 = Enable IP address translation (default).</p> <p>When enabled, the gateway compares the source IP address of the first incoming packet, to the remote IP address stated in the opening of the channel. If the two IP addresses don't match, the NAT mechanism is activated. Consequently, the remote IP address of the outgoing stream is replaced by the source IP address of the first incoming packet.</p> <p><b>Note:</b> The NAT mechanism must be enabled for this parameter to take effect (DisableNAT = 0).</p>
<b>EnableUDPPortTranslation</b>	<p>0 = Disable UDP port translation (default).</p> <p>1 = Enable UDP port translation.</p> <p>When enabled, the gateway compares the source UDP port of the first incoming packet, to the remote UDP port stated in the opening of the channel. If the two UDP ports don't match, the NAT mechanism is activated. Consequently, the remote UDP port of the outgoing stream is replaced by the source UDP port of the first incoming packet.</p> <p><b>Note:</b> The NAT mechanism and the IP address translation must be enabled for this parameter to take effect (DisableNAT = 0, EnableIPAddrTranslation = 1).</p>
<b>StaticNATIP</b> [NAT IP Address]	<p>Static NAT IP address.</p> <p>Global gateway IP address. Define if static Network Address Translation (NAT) device is located between the gateway and the Internet.</p>
<b>SyslogServerIP</b> [Syslog Server IP Address]	<p>IP address (in dotted format notation) of the computer you are using to run the Syslog Server.</p> <p>The Syslog Server is an application designed to collect the logs and error messages generated by the VoIP gateway.</p> <p><b>Note:</b> The default UDP Syslog port is 514.</p> <p>For information on the Syslog server, refer to Section 9.2 on page 165.</p>
<b>EnableSyslog</b> [Enable Syslog]	<p>1 = Send the logs and error message generated by the gateway to the Syslog Server. If you select 1, you must enter an IP address in the parameter SyslogServerIP.</p> <p>0 = Logs and errors are not sent to the Syslog Server (default).</p> <p><b>Note 1:</b> Syslog messages may increase the network traffic.</p> <p><b>Note 2:</b> To configure the Syslog logging levels use the parameter 'GwDebugLevel'.</p>

**Table 6-1: Basic, Logging, Web and RADIUS Parameters (continues on pages 90 to 97)**

<i>ini</i> File Field Name Web Parameter Name	Valid Range and Description
<b>BaseUDPport</b> [RTP Base UDP Port]	Lower boundary of UDP port used for RTP, RTCP (Real-Time Control Protocol) (RTP port + 1) and T.38 (RTP port + 2). The upper boundary is the Base UDP Port + 10 * (number of gateway's channels). The range of possible UDP ports is 4000 to 64000. The default base UDP port is 6000.  For example: If the Base UDP Port is set to 6000 (the default) then: The first channel uses the following ports: RTP 6000, RTCP 6001 and T.38 6002, the second channel uses: RTP 6010, RTCP 6011 and T.38 6012, etc.  <b>Note:</b> If RTP Base UDP Port is not a factor of 10, the following message is generated: "invalid local RTP port". For detailed information on the default RTP/RTCP/T.38 port allocation, refer to Section C.3 on page 202.
<b>IPDiffServ</b> [RTP IP Diff. Serv]	Diff Serv Code Point (DSCP) value that is assigned to the RTP packets. The DSCP value is used by DiffServ compatible routers to prioritize how packets are sent. By prioritizing packets, the DiffServ routers can minimize the transmission delays for time sensitive packets such as VoIP packets. The valid range is 0 to 63. The default value is 0. <b>Note:</b> The parameter IPDiffServ mustn't be used simultaneously with the parameters IPTOS and IPPrecedence.
<b>IPPrecedence</b> [RTP IP Precedence]	Value that is assigned to IP Type Of Service (TOS) field in the IP header for all RTP packets sent by the VoIP gateway. The valid range is 0 to 15. The default value is 0. <b>Note:</b> The parameters IPTOS and IPPrecedence mustn't be used simultaneously with the parameter IPDiffServ.
<b>IPTOS</b> [RTP IP TOS]	Value that is assigned to the IP Precedence field in the IP header for all RTP packets sent by the VoIP gateway. The valid range is 0 to 7. The default value is 0. <b>Note:</b> The parameters IPTOS and IPPrecedence mustn't be used simultaneously with the parameter IPDiffServ.
<b>MaxEchoCancellerLength</b> and <b>EchoCancellerLength</b>  <b>Note:</b> Both parameters must be set to the same value.	Maximum Echo Canceller Length in msec: 0 = Internal decision to keep max channel capacity (currently 32 msec) 4 = 32 msec 11 = 64 msec 22 = 128 msec The default value is 0.  <b>Note 1:</b> When set to 64 msec or more, the number of available gateway channels is reduced (by a factor of 5/6). For example: Gateway with 8 E1 spans capacity is reduced to 6 spans (180 channels), while gateway with 8 T1 spans capacity remains the same (192 channels). <b>Note 2:</b> The gateway must be reset after the value of 'MaxEchoCancellerLength' is changed.
<b>ECHybridLoss</b>	Sets the four wire to two wire worst case Hybrid loss, the ratio between the signal level sent to the hybrid and the echo level returning from the hybrid. 0 = 6 dB (default) 1 = 9 dB 2 = 0 dB 3 = 3 dB

Table 6-1: Basic, Logging, Web and RADIUS Parameters (continues on pages 90 to 97)

<i>ini</i> File Field Name Web Parameter Name	Valid Range and Description
<b>GwDebugLevel</b> [Debug Level]	Defines the Syslog logging level (usually set to 5 if debug traces are needed).  0 = Debug is disabled (default) 1 = Flow debugging is enabled 2 = Flow and board interface debugging are enabled 3 = Flow, board interface and stack interface debugging are enabled 4 = Flow, board interface, stack interface and session manager debugging are enabled 5 = Flow, board interface, stack interface, session manager and board interface expanded debugging are enabled
<b>CDRReportLevel</b> [CDR Report Level]	0 = CDR is not used 1 = Call Detail Record is sent to the Syslog server at the end of each call. 2 = CDR report is send to Syslog at start and at the end of each call The CDR Syslog message complies with RFC 3161 and is identified by: Facility = 17 (local1) and Severity = 6 (Informational)  <b>Note:</b> this parameter replaces the previous "EnableCDR" parameter
<b>CDRSyslogServerIP</b> [CDR Server IP Address]	Defines the destination IP address for CDR logs.  The default value is a null string that causes the CDR messages to be sent with all Syslog messages. <b>Note:</b> The CDR messages are sent to UDP port 514 (default Syslog port).
<b>HeartBeatDestIP</b>	Destination IP address (in dotted format notation) to which the gateway sends proprietary UDP 'ping' packets. The default IP address is 0.0.0.0.
<b>HeartBeatDestPort</b>	Destination UDP port to which the heartbeat packets are sent. The range is 0 to 64000. The default is 0.
<b>HeartBeatIntervalmsec</b>	Delay (in msec) between consecutive heartbeat packets. 10 = 100000. -1 = disabled (default).
<b>NTPServerIP</b> [NTP Server IP Address]	IP address (in dotted format notation) of the NTP server. The default IP address is 0.0.0.0 (the internal NTP client is disabled). For information on NTP support, refer to Section 5.9.1.3 on page 63.
<b>NTPServerUTCOffset</b> [NTP UTC Offset]	Defines the UTC (Universal Time Coordinate) offset (in seconds) from the NTP server. The default offset is 0. The offset range is -43200 to 43200 seconds.
<b>NTPUpdateInterval</b> [NTP Update Interval]	Defines the time interval (in seconds) the NTP client requests for a time update. The default interval is 86400 seconds (24 hours). The range is 0 to 214783647 seconds. <b>Note:</b> It isn't recommended to be set beyond one month (2592000 seconds).
<b>EnableRAI</b>	0 = Disable RAI (Resource Available Indication) service (default). 1 = Enable RAI service.  If RAI is enabled, an SNMP 'acBoardCallResourcesAlarm' Alarm Trap is sent if gateway resources fall below a predefined (configurable) threshold.
<b>RAIHighThreshold</b>	High Threshold (in percentage) that defines the gateway's busy endpoints. The range is 0 to 100. The default value is 90%.  When the percentage of the gateway's busy endpoints exceeds the value configured in High Threshold, the gateway sends an SNMP 'acBoardCallResourcesAlarm' Alarm Trap with a 'major' Alarm Status. <b>Note:</b> The gateway's available Resources are calculated by dividing the number of busy endpoints by the total number of available gateway endpoints.

**Table 6-1: Basic, Logging, Web and RADIUS Parameters (continues on pages 90 to 97)**

<i>ini</i> File Field Name Web Parameter Name	Valid Range and Description
<b>RAILowThreshold</b>	Low Threshold (in percentage) that defines the gateway's busy endpoints. The range is 0 to 100. The default value is 90%.  When the percentage of the gateway's busy endpoints falls below the value defined in Low Threshold, the gateway sends an SNMP 'acBoardCallResourcesAlarm' Alarm Trap with a 'cleared' Alarm Status.
<b>RAILoopTime</b>	Time interval (in seconds) that the gateway checks for resource availability. The default is 10 seconds.
<b>Disconnect Supervision Parameters</b>	
<b>DisconnectOnBrokenConnection</b> [Disconnect on Broken Connection]	0 = Don't release the call. 1 = Call is released If RTP packets are not received for a predefined timeout (default).  <b>Note 1:</b> If enabled, the timeout is set by the parameter 'BrokenConnectionEventTimeout', in 100 msec resolution. The default timeout is 10 seconds: (BrokenConnectionEventTimeout =100). <b>Note 2:</b> This feature is applicable only if RTP session is used without Silence Compression. If Silence Compression is enabled, the Gateway doesn't detect that the RTP connection is broken. <b>Note 3:</b> During a call, if the source IP address (from where the RTP packets were sent) is changed without notifying the Gateway, the Gateway will filter these RTP packets. To overcome this issue, set 'DisconnectOnBrokenConnection=0'; the Gateway doesn't detect RTP packets arriving from the original source IP address, and will switch (after 300 msec) to the RTP packets arriving from the new source IP address.
<b>BrokenConnectionEventTimeout</b> [Broken Connection Timeout]	The amount of time (in 100 msec units) an RTP packets isn't received, after which a call is disconnected. The valid range is 1 to 1000. The default value is 100 (10 seconds). <b>Note 1:</b> Applicable only if 'DisconnectOnBrokenConnection = 1'. <b>Note 2:</b> Currently this feature works only if Silence Suppression is disabled.
<b>EnableSilenceDisconnect</b> [Disconnect Call on Silence Detection]	1 = The gateway disconnect calls in which silence occurs in both (call) directions for more than 120 seconds. 0 = Call is not disconnected when silence is detected (default).  The silence duration can be set by the 'FarEndDisconnectSilencePeriod' parameter (default 120). <b>Note:</b> To activate this feature set: 'EnableSilenceCompression' to 1 and 'FarEndDisconnectSilenceMethod' to 1.
<b>FarEndDisconnectSilencePeriod</b> [Silence Detection Period]	Duration of silence period (in seconds) prior to call disconnection. The range is 10 to 28800 (8 hours). The default is 120 seconds. Applicable to gateways, that use DSP templates 2 or 3.
<b>FarEndDisconnectSilenceMethod</b> [Silence Detection Method]	Silence detection method. 0 (None) = Silence detection option is disabled. 1 (Packets Count) = According to packet count. 2 (Voice/Energy Detectors) = N/A. 3 (All) = N/A.
<b>FarEndDisconnectSilenceThreshold</b>	Threshold of the packet count (in percents), below which is considered silence by the media gateway. The valid range is 1 to 100. The default is 8%. <b>Note:</b> Applicable only if silence is detected according to packet count (FarEndDisconnectSilenceMethod = 1).
<b>Web-Related Parameters</b>	
<b>DisableWebTask</b>	0 = Enable Web management (default) 1 = Disable Web management

Table 6-1: Basic, Logging, Web and RADIUS Parameters (continues on pages 90 to 97)

<i>ini</i> File Field Name Web Parameter Name	Valid Range and Description
<b>ResetWebPassword</b>	Allows resetting to default of Web password to: Username: "Admin" Password: "Admin"
<b>DisableWebConfig</b>	0 = Enable changing parameters from Web (default) 1 = Operate Web Server in "read only" mode
<b>HTTPport</b>	HTTP port used for Web management (default = 80)
<b>Customizing the Web Appearance Parameters</b> For detailed information on customizing the Web Interface, refer to <a href="#">Appendix F</a> on page 207.	
<b>UseProductName</b>	0 = Disabled (default). 1 = Enabled. If enabled, the 'UserProductNane' text string is displayed instead of the default product name.
<b>UserProductName</b>	Text string that replaces the product name. The default is "Mediant 2000". The string can be up to 29 characters.
<b>UseWebLogo</b>	0 = Logo image is used (default). 1 = Text string is used instead of a logo image. If enabled, AudioCodes' default logo (or any other logo defined by the 'LogoFileName' parameter) is replaced with a text string defined by the 'WebLogoText' parameter.
<b>WebLogoText</b>	Text string that replaces the logo image. The string can be up to 15 characters.
<b>LogoWidth</b>	Width (in pixels) of the logo image. <b>Note:</b> The optimal setting depends on the resolution settings. The default value is 441, which is the width of AudioCodes' displayed logo.
<b>LogoFileName</b>	Name of the image file containing the user's logo. File name can be up to 47 characters. The logo file name can be used to replace AudioCodes' default Web logo with a User defined logo. Use a gif, jpeg or jpg image file.
<b>BkgImageFileName</b>	Name of the file containing the user's background image. File name can be up to 47 characters. The background file can be used to replace AudioCodes' default background image with a User defined background. Use a gif, jpeg or jpg image file.
<b>RADIUS-Related Parameters</b> For detailed information on the supported RADIUS attributes, refer to Section <a href="#">K.7</a> on page 251.	
<b>EnableRADIUS</b> [Enable RADIUS]	0 = RADIUS application is disabled (default). 1 = RADIUS application is enabled.
<b>AAAIindications</b> [AAA Indications]	0 = No indications (default). 3 = Accounting only
<b>MaxRADIUSSessions</b> [Max. RADIUS Sessions]	Number of concurrent calls that can communicate with the RADIUS server (optional). The valid range is 0 to 240. The default value is 240.
<b>SharedSecret</b> [Shared Secret]	"Secret" used to authenticate the gateway to the RADIUS server. It should be a cryptographically strong password.
<b>RADIUSRetransmission</b> [RADIUS Max. Retransmissions]	Number of retransmission retries. The valid range is 1 to 10. The default value is 3.
<b>RadiusTO</b>	The interval between each retry (measured in seconds). The valid range is 1 to 30. The default value is 10.
<b>RADIUSAuthServerIP</b> [RADIUS Authentication Server IP Address]	IP address of Authentication and Authorization server.

**Table 6-1: Basic, Logging, Web and RADIUS Parameters (continues on pages 90 to 97)**

<i>ini</i> File Field Name Web Parameter Name *	Valid Range and Description
<b>RADIUSAuthPort</b> [RADIUS Authentication Port]	Port number of Authentication and Authorization server. The default value is 1645.
<b>RADIUSAccServerIP</b> [RADIUS Accounting Server IP Address]	IP address of accounting server.
<b>RADIUSAccPort</b> [RADIUS Accounting Port]	Port number of Radius accounting server. The default value is 1646.
<b>RadiusAccountingType</b> [RADIUS Accounting Type]	Determines when a RADIUS accounting report is issued. 0 = At the Release of the call only (default). 1 = At the Connect and Release of the call. 2 = At the Setup and Release of the call.
<b>BootP and TFTP Parameters</b>	
<b>IniFileURL</b>	Specifies the name of the <i>ini</i> file and the location of the TFTP server from which the gateway loads the <i>ini</i> and configuration files. For example: tftp://192.168.0.1/filename tftp://192.10.77.13/config<MAC> <b>Note:</b> The optional string "<MAC>" is replaced with the gateway's MAC address. Therefore, the gateway requests an <i>ini</i> file name that contains its MAC address. This option enables loading different configurations for specific gateways.
<b>CmpFileURL</b>	Specifies the name of the <i>cmp</i> file and the location of the TFTP server from which the gateway loads a new <i>cmp</i> file and updates itself. For example: tftp://192.168.0.1/filename <b>Note 1:</b> When this parameter is set in the <i>ini</i> file, the gateway <u>always</u> loads the <i>cmp</i> file after it is reset. <b>Note 2:</b> The version of the loaded <i>cmp</i> file isn't checked.
The BootP parameters are special "Hidden" parameters. Once defined and saved in the flash memory, they are used even if they don't appear in the <i>ini</i> file.	
<b>BootPRetries</b>	BootP retries. Sets the number of BootP requests the device sends during start-up. The device stops sending BootP requests when either BootP reply is received or Number of Retries is reached. 1 = 1 BootP retry, 1 second. 2 = 2 BootP retries, 3 second. 3 = 3 BootP retries, 6 second (default). 4 = 10 BootP retries, 30 second. 5 = 20 BootP retries, 60 second. 6 = 40 BootP retries, 120 second. 7 = 100 BootP retries, 300 second. 15 = BootP retries indefinitely. <b>Note:</b> This parameter only takes effect from the next reset of the device.
<b>BootPSelectiveEnable</b>	Enables the Selective BootP mechanism. 1 = Enabled. 0 = Disabled (default).  The Selective BootP mechanism enables the gateway's integral BootP client to filter unsolicited BootP/DHCP replies (accepts only BootP replies that contain the text "AUDC" in the vendor specific information field). This option is useful in environments where enterprise BootP/DHCP servers provide undesired responses to the gateway's BootP requests. <b>Note:</b> When working with DHCP (EnableDHCP=1) the selective BootP feature must be disabled.

Table 6-1: Basic, Logging, Web and RADIUS Parameters (continues on pages 90 to 97)

<i>ini</i> File Field Name Web Parameter Name	Valid Range and Description
<b>BootPDelay</b>	<p>The interval between the device's startup and the first BootP/DHCP request that is issued by the device.</p> <p>1 = 1 second (default).            2 = 3 second.            3 = 6 second.            4 = 30 second.            5 = 60 second.</p> <p><b>Note:</b> This parameter only takes effect from the next reset of the device.</p>
<b>ExtBootPReqEnable</b>	<p>0 = Disable (default).            1 = Enable extended information to be sent in BootP request.</p> <p>If enabled, the device uses the vendor specific information field in the BootP request to provide device-related initial startup information such as board type, current IP address, software version, etc. For a full list of the vendor specific Information fields, refer to Section 10.3 on page 169.</p> <p>The BootP/TFTP configuration utility displays this information in the 'Client Info' column (refer to Figure B-1).</p> <p><b>Note:</b> This option is not available on DHCP servers.</p>

## 6.7 SNMP Parameters



**Note:** In Table 6-2, parameters in brackets are the format in the Embedded Web Server \*.

Table 6-2: SNMP Parameter (continues on pages 98 to 99)

<i>ini</i> File Field Name Web Parameter Name *	Valid Range and Description
<b>DisableSNMP</b> [Enable SNMP]	0 = SNMP is enabled (default). 1 = SNMP is disabled and no traps are sent.
<b>SNMPPort</b>	The device's local UDP port used for SNMP Get/Set commands. The range is 100 to 3999. The default port is 161.
<b>SNMPTrustedMGR_x</b>	Up to five IP addresses of remote trusted SNMP managers from which the SNMP agent accepts and processes get and set requests. <b>Note 1:</b> If no values are assigned to these parameters any manager can access the device. <b>Note 2:</b> Trusted managers can work with <i>all</i> community strings.
<b>SNMP Trap Parameters</b>	
<b>SNMPManagerTableIP_x</b> [SNMP Managers Table]	Up to five IP addresses of remote hosts that are used as SNMP Managers. The device sends SNMP traps to these IP addresses. Enter the IP address in dotted format notation, for example 108.10.1.255. <b>Note:</b> The first entry (out of the five) replaces the obsolete parameter SNMPManagerIP.
<b>SNMPManagerTrapPort_x</b> [SNMP Managers Table]	Up to five parameters used to define the Port numbers of the remote SNMP Managers. The device sends SNMP traps to these ports. <b>Note:</b> The first entry (out of the five) replaces the obsolete parameter SNMPTrapPort. The default SNMP trap port is 162 The SNMP trap port must be between 100 to 4000.
<b>SNMPManagerIsUsed_x</b> [SNMP Managers Table]	Up to five parameters, each determines the <b>validity</b> of the parameters (IP address and port number) of the corresponding SNMP Manager used to receive SNMP traps. 0 = Disabled (default) 1 = Enabled
<b>SNMPManagerTrapSendingEnable_x</b> [SNMP Managers Table]	Up to five parameters, each determines the activation/deactivation of sending traps to the corresponding SNMP Manager. 0 = Sending is disabled 1 = Sending is enabled (default)
<b>SNMPManagerIP</b>  <b>Note:</b> Obsolete parameter, use SNMPManagerTableIP_x instead.	IP address (in dotted format notation) for the computer that is used as the <i>first</i> SNMP Manager. The SNMP Manager is a device that is used for receiving SNMP Traps.  <b>Note 1:</b> To enable the device to send SNMP Traps, set the <i>ini</i> file parameter SNMPManagerIsUsed to 1. <b>Note 2:</b> If you want to use more than one SNMP manger, ignore this parameter and use the parameters 'SNMPManagerTableIP_x' instead.
<b>SNMP Community String Parameters</b>	
<b>SNMPReadOnlyCommunityString_x</b>	Read-only community string (up to 19 chars). The default string is "public".
<b>SNMPReadWriteCommunityString_x</b>	Read-write community string (up to 19 chars). The default string is "private".
<b>SNMPTrapCommunityString_x</b>	Community string used in traps (up to 19 chars). The default string is "trapuser".

Table 6-2: SNMP Parameter (continues on pages 98 to 99)

<b><i>ini</i> File Field Name Web Parameter Name</b>	<b>Valid Range and Description</b>
<b>SetCommunityString</b> <b>Note:</b> Obsolete parameter, use SNMPReadWriteCommunityString_x instead.	SNMP community string (up to 19 chars). Default community string for read "public", for set & get "private".

## 6.8 SIP Configuration Parameters



**Note:** In Table 6-3, parameters in brackets are the format in the Embedded Web Server \*.

**Table 6-3: SIP Configuration Parameters (continues on pages 100 to 110)**

<i>ini</i> File Field Name Web Parameter Name *	Valid Range and Description
<b>ControlIPDiffServ</b> [Signaling DiffServ]	Defines the value of the 'DiffServ' field in the IP header for SIP messages. The valid range is 0 to 63. The default value is 0.
<b>SIPGatewayName</b> [Gateway Name]	Use this parameter to assign a name to the device (For example: 'gateway1.com'). Ensure that the name you choose is the one that the Proxy is configured with to identify your media gateway. <b>Note:</b> If specified, the gateway Name is used as the host part of the SIP URL, in the 'From' header. If not specified, the gateway IP address is used instead (default).
<b>IsProxyUsed</b> [Enable Proxy]	0 = Proxy isn't used, the internal routing table is used instead (default). 1 = Proxy is used.
<b>ProxyIP</b> [Proxy IP Address]	IP address (and optionally port number) of the primary Proxy server you are using. Enter the IP address as FQDN or in dotted format notation (for example 201.10.8.1). You can also specify the selected port in the format: <IP Address>:<port>.  This parameter is applicable only if you select 'Yes' in the 'Is Proxy Used' field. If you enable Proxy Redundancy (by setting EnableProxyKeepAlive=1), the gateway can function with up to three Proxy servers. If there is no response from the primary Proxy, the gateway tries to communicate with the redundant Proxies. When a redundant Proxy is found, the gateway either continues working with it until the next failure occurs or reverts to the primary Proxy (refer to the 'Redundancy Mode' parameter). If none of the Proxy servers respond, the gateway goes over the list again.  The gateway also provides real time switching (hotswap mode), between the primary and redundant proxies ('IsProxyHotSwap=1'). If the first Proxy doesn't respond to Invite message, the same Invite message is immediately sent to the second Proxy. <b>Note 1:</b> If 'EnableProxyKeepAlive=1', the gateway monitors the connection with the Proxies by using keep-alive messages ("OPTIONS"). <b>Note 2:</b> To use Proxy Redundancy, you must specify one or more redundant Proxies using multiple 'ProxyIP= <IP address>' definitions. <b>Note 3:</b> When port number is specified (e.g., domain.com:5080), DNS SRV queries aren't performed, even if 'EnableProxySRVQuery' is set to 1.
<b>ProxyIP</b> <b>ProxyIP</b> <b>ProxyIP</b> [Redundant Proxy IP Address]	IP addresses of the redundant Proxies you are using. Enter the IP address as FQDN or in dotted format notation (for example 192.10.1.255). You can also specify the selected port in the format: <IP Address>:<port>.  <b>Note 1:</b> This parameter is available only if you select "Yes" in the 'Is Proxy Used' field. <b>Note 2:</b> When port number is specified, DNS SRV queries aren't performed, even if 'EnableProxySRVQuery' is set to 1. <b>ini file note:</b> The IP addresses of the redundant Proxies are defined by the second, third and fourth repetition of the <i>ini</i> file parameter 'ProxyIP'.
<b>ProxyName</b> [Proxy Name]	Home Proxy Domain Name. If specified, the name is used as Request-URI in REGISTER, INVITE and other SIP messages. If the proxy name isn't specified, the Proxy IP address is used instead.

Table 6-3: SIP Configuration Parameters (continues on pages 100 to 110)

<i>ini</i> File Field Name Web Parameter Name *	Valid Range and Description
<b>EnableProxySRVQuery</b> [Enable Proxy SRV Queries]	Enables the use of DNS Service Record (SRV) queries to discover Proxy servers. 0 = Disabled (default). 1 = Enabled.  If enabled and the Proxy IP address parameter contains a domain name without port definition (e.g., ProxyIP = domain.com), an SRV query is performed. The SRV query returns up to four Proxy host names and their weights. The gateway then performs DNS A-record queries for each Proxy host name (according to the received weights) to locate up to four Proxy IP addresses. Therefore, if the first SRV query returns two domain names, and the A-record queries return 2 IP addresses each, no more searches are performed. If the Proxy IP address parameter contains a domain name with port definition (e.g., ProxyIP = domain.com:5080), the gateway performs a regular DNS A-record query. <b>Note:</b> This mechanism is applicable only if 'EnableProxyKeepAlive = 1'.
<b>AlwaysSendToProxy</b> [Always Use Proxy]	0 = Use standard SIP routing rules (default) 1 = All SIP messages and Responses are sent to Proxy server <b>Note:</b> Applicable only if Proxy server is used.
<b>SendInviteToProxy</b> [Send All Invite to Proxy]	0 = INVITE messages, generated as a result of Transfer or Redirect, are sent directly to the URL (according to the refer-to header in the REFER message or contact header in 30x response) (default). 1 = All INVITE messages, including those generated as a result of Transfer or Redirect are sent to Proxy. <b>Note:</b> Applicable only if Proxy server is used and "AlwaysSendtoProxy=0".
<b>PreferRouteTable</b> [Prefer Routing Table]	Determines if the local Tel to IP routing table takes precedence over a Proxy for routing calls. 0 = Only Proxy is used to route calls (default). 1 = The Proxy checks the 'Destination IP Address' field in the 'Tel to IP Routing' table for a match with the outgoing call. Only if a match is not found, a Proxy is used. <b>Note:</b> Applicable only if Proxy is not always used ('AlwaysSendToProxy' = 0, 'SendInviteToProxy' = 0).
<b>EnableProxyKeepAlive</b> [Enable Proxy Keep Alive]	0 = Disable (default) 1 = Keep alive with Proxy, by sending "OPTIONS" SIP message every "ProxyKeepAliveTime". <b>Note:</b> This parameter must be enabled when Proxy redundancy is used.
<b>ProxyKeepAliveTime</b> [Proxy Keep Alive Time]	Defines the Proxy keep-alive time interval (in seconds) between OPTIONS messages. The default value is 60 seconds.
<b>UseGatewayNameForOptions</b> [Use Gateway Name for OPTIONS]	0 = Use the gateway's IP address in keep-alive OPTIONS messages (default). 1 = Use 'GatewayName' in keep-alive OPTIONS messages. The OPTIONS Request-URI host part contains either the gateway's IP address or a string defined by the parameter 'Gatewayname'. The gateway uses the OPTIONS request as a keep-alive message to its primary and redundant Proxies.
<b>IsProxyHotSwap</b> [Enable Proxy Hotswap]	Enable Proxy Hot Swap redundancy mode. 0 = Disabled (default) 1 = Enabled If Hot Swap is enabled, SIP INVITE message is first sent to the primary Proxy server. If there is no response from the primary Proxy server for "ProxyHotSwapRtx" retransmissions, the INVITE message is resent to the redundant Proxy server.
<b>ProxyHotSwapRtx</b> [Number of RTX before Hotswap]	Number of retransmitted INVITE messages before call is routed (hot swap) to another Proxy Range: 1-30 The default is 3. <b>Note:</b> This parameter is also used for alternative routing using the Tel to IP Routing table. If a domain name in the routing table is resolved into 2 IP addresses, and if there is no response for 'ProxyHotSwapRtx' retransmissions to the Invite message that is sent to the first IP address, the gateway immediately initiates a call to the second IP address.

**Table 6-3: SIP Configuration Parameters (continues on pages 100 to 110)**

<i>ini</i> File Field Name Web Parameter Name *	Valid Range and Description
<b>ProxyRedundancyMode</b> [Redundancy Mode]	0 = Parking mode: gateway continues working with the last active Proxy until the next failure. (default) 1 = Homing mode: gateway always tries to work with the primary Proxy server (switches back to the primary Proxy whenever it is available).  <b>Note:</b> To use ProxyRedundancyMode, enable Keep-alive with Proxy option (EnableProxyKeepAlive=1).
<b>IsTrustedProxy</b> [Is Proxy Trusted]	This parameter isn't applicable and must always be set to 1. The parameter 'AssertedIdMode' should be used instead.
<b>IsFallbackUsed</b> [Enable Fallback to Routing Table]	0 = Gateway fallback is not used (default). 1 = Internal Telephone to IP Routing table is used when Proxy servers are not available. When the gateway falls back to the internal Telephone to IP Routing table, the gateway continues scanning for a Proxy. When the gateway finds an active Proxy, it switches from internal routing back to Proxy routing. <b>Note:</b> To enable the redundant Proxies mechanism set 'EnableProxyKeepAlive' to 1.
<b>UserName</b> [User Name]	Username used for Registration and for Basic/Digest authentication process with Proxy / Registrar. Parameter doesn't have a default value (empty string).
<b>Password</b> [Password]	Password used for Basic/Digest authentication process with Proxy / Registrar. The default is "Default_Passwd".
<b>Cnonce</b> [Cnonce]	String used by the SIP Server and client to provide mutual authentication (free format, i.e., "Cnonce = 0a4f113b"). The default is "Default_Cnonce".
<b>IsRegisterNeeded</b> [Enable Registration]	0 = Gateway does not register to Proxy/Registrar (default) 1 = Gateway registers to Proxy/Registrar at power up
<b>RegistrarIP</b> [Registrar IP Address]	IP address of Registrar server (optional). If not specified, the gateway registers to Proxy server.
<b>RegistrarName</b> [Registrar Name]	Registrar Domain Name. If specified, the name is used as Request-URI in Register messages. If isn't specified (default), the Registrar IP address or Proxy name or Proxy IP address is used instead.
<b>GWRegistrationName</b> [Gateway Registration Name]	Defines the user name that is used in From and To headers of Register messages. If 'GWRegistrationName' isn't specified (default), the 'Username' parameter is used instead.
<b>RegistrationTime</b> [Registration Time]	Registration expired timeout (seconds). The value is used in "Expires = " header. Typically a value of 3600 is assigned, for one hour registration. The gateway resumes registration when half the defined timeout period expires.
<b>RegistrationTimeDivider</b> [Re-registration Timing (%)]	Defines the re-registration timing (in percentage). The timing is a percentage of the re-register timing set by the Registration server. The valid range is 50 to 100. The default value is 50. For example: If 'RegistrationTimeDivider = 70' (%) and Registration Expires time = 3600, the gateway resends its registration request after 3600 x 70% = 2520 sec.
<b>RegistrationRetryTime</b> [Registration Retry Time]	Defines the time period (in seconds) after which a Registration request is resent if registration fails with 4xx, or there is no response from the Proxy/Registrar. The default is 30 seconds. The range is 10 to 3600.
<b>PrackMode</b> [PRACK Mode]	PRACK mechanism mode for 1XX reliable responses: 0 = Disabled 1 = Supported (default) 2 = Required  <b>Note 1:</b> The Supported and Required headers contain the "100rel" parameter. <b>Note 2:</b> The Mediant 2000 sends PRACK message if 180/183 response is received with "100rel" in the Supported or the Required headers.

Table 6-3: SIP Configuration Parameters (continues on pages 100 to 110)

<b>ini File Field Name Web Parameter Name *</b>	<b>Valid Range and Description</b>
<b>AssertedIdMode</b> [Asserted Identity Mode]	0 = None (default). 1 = P-asserted. 2 = P-preferred.  The Asserted ID mode defines the header that is used in the generated INVITE request. The header also depends on the calling Privacy: allowed or restricted. The P-asserted (or P-preferred) headers are used to present the originating party's Caller ID. The Caller ID is composed of a Calling Number and (optionally) a Calling Name. P-asserted (or P-preferred) headers are used together with the Privacy header. If Caller ID is restricted the "Privacy: id" is included. Otherwise, for allowed Caller ID the "Privacy: none" is used. If Caller ID (received from PSTN) is restricted, the From header is set to <anonymous@anonymous.invalid>.
<b>EnableRPIheader</b> [Enable Remote Party ID]	0 = Disable (default) 1 = RPI (Remote-Party-ID) headers are generated in SIP INVITE message for both called and calling numbers.
<b>SIPDestinationPort</b> [SIP Destination Port]	SIP UDP destination port for sending SIP messages. The default port is 5060.
<b>LocalSIPPort</b> [SIP Local Port]	Local UDP port used to receive SIP messages. The default port is 5060.
<b>IsUserPhone</b> [Use "user=phone" in SIP URL]	0 = Doesn't use "user=phone" string in SIP URL. 1 = "user=phone" string is part of the SIP URL (default).
<b>IsUserPhoneInFrom</b> [Use "user=phone" in From header]	0 = Doesn't use ";user=phone" string in From header (default). 1 = ";user=phone" string is part of the From header.
<b>EnablePtime</b>	0 = Remove the ptime header from SDP. 1 = Include the ptime header in SDP (default).

**Table 6-3: SIP Configuration Parameters (continues on pages 100 to 110)**

<i>ini</i> File Field Name Web Parameter Name *	Valid Range and Description
<b>CoderName</b> [Coders]	<p>CoderName = Coder,ptime (can appear up to 5 times) The following coder names can be selected:</p> <ul style="list-style-type: none"> <li>g711Alaw64k – G.711 A-law.</li> <li>g711Ulaw64k – G.711 <math>\mu</math>-law.</li> <li>g7231 – G.723.1 6.3 kbps (default).</li> <li>g7231r53 – G.723 5.3 kbps.</li> <li>g726 – G.726 ADPCM 32 kbps (Payload Type = 2).</li> <li>g729 – G.729A.</li> <li>NetCoder6_4 – NetCoder 6.4 kbps.</li> <li>NetCoder7_2 – NetCoder 7.2 kbps.</li> <li>NetCoder8 – NetCoder 8.0 kbps.</li> <li>NetCoder8_8 – NetCoder 8.8 kbps.</li> <li>Transparent – Transparent coder.</li> <li>EvrC – EVRC coder.</li> <li>Amr – AMR coder.</li> </ul> <p><b>Note:</b> The coder name is case-sensitive.</p> <p>The RTP packetization period (ptime, in msec) depends on the selected Coder name, and can have the following values:</p> <ul style="list-style-type: none"> <li>g711 family – 10,20,30,40,50,60,80,100,120 (default=20).</li> <li>g729 – 10,20,30,40,50,60,80,100 (default=20).</li> <li>g723 family – 30,60,90,120 (default = 30).</li> <li>G.726 – 10, 20, 40, 60, 80, 100, 120 (default=20).</li> <li>NetCoder family – 20, 40, 60, 80, 100, 120 (default=20).</li> <li>EVRC – 20, 40, 60, 80, 100 (default=20).</li> <li>AMR – 20 only.</li> <li>Transparent – 20, 40, 60, 80, 100, 120 (default=20)</li> </ul> <p><b>Note 1:</b> If not specified, the ptime gets a default value.  <b>Note 2:</b> Each coder should appear only once.  <b>Note 3:</b> The ptime specifies the maximum packetization time the gateway can receive.  <b>Note 4:</b> G.729B is supported if the coder G.729 is selected and 'EnableSilenceCompression' equals 1 or 2.  <b>Note 5:</b> The selected rate of the AMR coder is set according to the parameter 'AMRSendRate'. The selected rate of the EVRC coder is set according to the parameter 'EVRCCRate'.  <b>Note 6:</b> The AMR coder is enabled only if 'DSPVersionTemplateNumber = 1'. When this DSP template is selected, the maximum number of channels is 160 instead of 240 (rounded to full E1/T1 trunk capacity 30/24 channels per trunk).  <b>Note 7:</b> The EVRC coder is enabled only if 'DSPVersionTemplateNumber = 2'. When this DSP template is selected, the maximum number of channels is 120 instead of 240 (rounded to full E1/T1 trunk capacity 30/24 channels per trunk). Note that the value of the parameter 'EnableRFC2658Interleaving' must be identical on both sides of the call.</p> <p>For example:                      CoderName = g711Alaw64k,20                      CoderName = g711Ulaw64k,40                      CoderName = g7231,90</p>
<b>AMRPayloadType</b>	<p>Determines the payload type that is used when the selected coder is set to one of the AMR coder variants. The valid range is 0 to 120. The default value is 64.</p>

Table 6-3: SIP Configuration Parameters (continues on pages 100 to 110)

<i>ini</i> File Field Name Web Parameter Name *	Valid Range and Description
<b>AMRSendRate</b>	Determines the selected rate for the AMR coder. This parameter is relevant only if AMR is included in the coder list ('CoderName'). 0 = AMR 4.75 kbps. 1 = AMR 5.15 kbps. 2 = AMR 5.90 kbps. 3 = AMR 6.70 kbps. 4 = AMR 7.40 kbps. 5 = AMR 7.95 kbps. 6 = AMR 10.2 kbps. 7 = AMR 12.2 kbps (default).
<b>EVRCRate</b>	Determines the selected rate for the EVRC coder. This parameter is relevant only if EVRC is included in the coder list ('CoderName'). 0 = Variable rate (default). 1 = 1 kbps 2 = 4 kbps 3 = 8 kbps
<b>EVRCPayloadType</b>	Determines the payload type that is used when the selected coder is set to 'Evcrc'. The valid range is 0 to 120. The default value is 60.
<b>TransparentPayloadType</b>	Specifies the payload type that is used when the selected coder is set to 'Transparent'. The valid range is 0 to 120. The default value is 56.
<b>DSPVersionTemplateNumber</b>	Determines the number of the DSP load. Each load has a different coder list, a different channel capacity and different supported features. 0 = G.711, G.726, G.723, G.729, Netcoder (default). 1 = AMR, G.711, G.726, G.723, G.729. 2 = EVRC, G.711, G.726, G.723, G.729.
<b>EnableRFC2658Interleaving</b>	When enabled, RTP packets include an interleaving byte for VBR coders. 0 = Disable (default). 1 = Enable. <b>Note:</b> This parameter is applicable only to EVRC coder.
<b>TransparentCoderOnDataCall</b>	0 = Only use coders from the coder list (default). 1 = Use transparent coder for data calls. The 'Transparent' coder can be used on data calls. When the gateway receives a Setup message from the ISDN with 'TransferCapabilities = data', it can initiate a call using the coder 'Transparent' (even if the coder is not included in the coder list). This option is a proprietary feature that requires the receiving gateway to include the 'Transparent' coder in its coder list.
<b>IsFaxUsed</b> [Fax Signaling Method]	Determines the SIP signaling method used to establish and convey a fax session after a fax is detected. 0 (No Fax) = No fax negotiation using SIP signaling (default). 1 (T.38 Relay) = Initiates T.38 fax relay. 2 (G.711 Transport) = Initiates fax using the coder G.711 A-law/ $\mu$ -law (if not previously selected) with adaptations (refer to note 1). <b>Note 1:</b> Fax adaptations: Echo Cancellor = On Silence Compression = Off Echo Cancellor Non-Linear Processor Mode = Off Dynamic Jitter Buffer Minimum Delay = 40 Dynamic Jitter Buffer Optimization Factor = 13 <b>Note 2:</b> If the gateway initiates a fax session using G.711 (option 2), a 'gpmid' attribute is added to the SDP in the following format: For A-law: 'a=gpmid:0 vbd=yes;ecan=on'. For $\mu$ -law: 'a=gpmid:8 vbd=yes;ecan=on'. <b>Note 3:</b> When 'IsFaxUsed' is set to 1 or 2, the parameter 'FaxTransportMode' is ignored.
<b>T38UseRTPPort</b>	Defines that the T.38 packets are sent / received using the same port as RTP packets. 0 = Use the RTP port +2 to send / receive T.38 packets (default). 1 = Use the same port as the RTP port to send / receive T.38 packets.

**Table 6-3: SIP Configuration Parameters (continues on pages 100 to 110)**

<b>ini File Field Name Web Parameter Name *</b>	<b>Valid Range and Description</b>
<b>CngDetectorMode</b> [CNG Detector Mode]	0 = Don't detect CNG (default) 2 = Detect CNG on caller side and start fax session (if IsFaxUsed=1) Usually T.38 fax session starts when the "preamble" signal is detected by the answering side. Some SIP gateways doesn't support the detection of this fax signal on the answering side, for these cases it is possible to configure the gateways to start the T.38 fax session when the CNG tone is detected by the originating side. However this mode is not recommended.
<b>DefaultReleaseCause</b> [Default Release Cause]	Default Release Cause (for IP to Tel calls), used when the gateway initiates a call release, and if an explicit matching cause for this release isn't found, a default release cause can be configured. The default release cause is described in the Q.931 notation, and translated to corresponding SIP equivalent response value  The default release cause is: NO_ROUTE_TO_DESTINATION (3). Other common values are: NO_CIRCUIT_AVAILABLE (34) or DESTINATION_OUT_OF_ORDER (27), etc. <b>Note:</b> The default release cause is described in the Q.931 notation, and is translated to corresponding SIP 40x or 50x value. For example: 404 for 3, 503 for 34 and 502 for 27. For mapping of SIP to Q.931 and Q.931 to SIP release causes, refer to <a href="#">Appendix I</a> on page 227.
<b>IPAlertTimeout</b> [Tel to IP No Answer Timeout]	Defines the time (in seconds) the gateway waits for a 200 OK response from the called party (IP side) after sending an Invite message. If the timer expires, the call is released. The valid range is 0 to 3600. The default value is 180.
<b>SipSessionExpires</b> [Session-Expires Time]	0 = Not activate (default) Timeout [seconds] for Keeping a "re-INVITE" message alive within a SIP session
<b>MINSE</b> [Minimum Session-Expires]	Defines the time (in seconds) that is used in the Min-SE header field. This field defines the minimum time that the user agent supports for session refresh. The valid range is 10 to 100000. The default value is 90.
<b>SIPMaxRtx</b> [SIP Maximum Rtx]	Number of UDP retransmissions of SIP messages. The range is 1 to 7. The default value is 7.
<b>SipT1Rtx</b> [SIP T1 Retransmission Timer (msec)]	The time interval (in msec) between the first transmission of a SIP message and the first retransmission of the same message. The default is 500. <b>Note:</b> The time interval between subsequent retransmissions of the same SIP message starts with SipT1Rtx and is multiplied by two until SipT2Rtx.  For example (assuming that SipT1Rtx = 500 and SipT2Rtx = 4000): The first retransmission is sent after 500 msec. The second retransmission is sent after 1000 (2*500) msec. The third retransmission is sent after 2000 (2*1000) msec. The fourth retransmission and subsequent retransmissions until SIPMaxRtx are sent after 4000 (2*2000) msec.
<b>SipT2Rtx</b> [SIP T2 Retransmission Timer (msec)]	The maximum interval (in msec) between retransmission of SIP messages. The default is 4000. <b>Note:</b> The time interval between subsequent retransmissions of the same SIP message starts with SipT1Rtx and is multiplied by two until SipT2Rtx.
<b>EnableEarlyMedia</b> [Enable Early Media]	0 = Early Media is disabled (default). 1 = Enable Early Media. If enabled, the Mediant 2000 gateway sends 183 Session Progress response with SDP (instead of 180 ringing), enabling the setup of the media stream prior to the answering of the call. Sending 183 response depends on the Progress Indicator. It is sent only if PI=1 or PI=8 was received in Proceeding or Alert PRI messages. For CAS gateways see the 'ProgressIndicator2IP' parameter. <b>Note:</b> Generally, this parameter is set to 1.
<b>EnableTransfer</b> [Enable Transfer]	0 = Call transfer is not allowed (default). 1 = The gateway responds to a Refer message with "Referred To" header to initiates a Call Transfer.

Table 6-3: SIP Configuration Parameters (continues on pages 100 to 110)

<i>ini</i> File Field Name Web Parameter Name *	Valid Range and Description
<b>XferPrefix</b> [Transfer Prefix]	Defined string that is added, as a prefix, to the transferred called number, when Refer/3xx message is received. <b>Note 1:</b> The number manipulation rules apply to the user part of the Refer-TO/Contact URL before it is sent in the Invite message. <b>Note 2:</b> The xferprefix parameter can be used to apply different manipulation rules to differentiate the transferred number from the original dialed number.
<b>EnableHold</b> [Enable Hold]	0 = Hold service disabled (default) 1 = Hold service is enabled, held tone is played to holding party.
<b>EnableForward</b> [Enable Call Forward]	0 = Disable call forward (default) 1 = Enable call forward service. The Mediant 2000 doesn't initiate call forward, it can only respond to call forward requests.
<b>UseSIPURIForDiversionHeader</b>	Sets the URI format in the Diversion header. 0 = "tel:" (default). 1 = "sip:".
<b>EnableCallWaiting</b> [Enable Call Waiting]	0 = Disabled (default) 1 = Enabled If enabled, when the gateway initiates a Tel to IP call to a destination that is busy, it plays a Call Waiting Ringback tone to the originator of the call. <b>Note 1:</b> The gateway's Call Progress Tones file must include a Call Waiting Ringback tone. <b>Note 2:</b> The EnableHold parameter must be enabled on the called side.
<b>RxDTMFOption</b> [Declare RFC 2833 in SDP]	Defines the supported Receive DTMF negotiation method. 0 = Don't declare RFC 2833 Telephony-event parameter in SDP 1 = n/a 2 = n/a 3 = Declare RFC 2833 "Telephony-event" parameter in SDP (default)  The gateway is designed to always be receptive to RFC 2833 DTMF relay packets. Therefore, it is always correct to include the "Telephony-event" parameter as a default in the SDP. However some gateways use the absence of the "telephony-event" from the SDP to decide to send DTMF digits inband using G.711 coder, if this is the case you can set "RxDTMFOption=0".
<b>TxDTMFOption</b> [DTMF RFC2833 Negotiation]	0 = No negotiation, DTMF digit is sent according to the "DTMFTransportType" parameter. 4 = Enable RFC 2833 payload type (PT) negotiation (default).  <b>Note 1:</b> This parameter is applicable only if "IsDTMFUsed=0" (out-of-band DTMF is not used) <b>Note 2:</b> If enabled, the gateway: <ul style="list-style-type: none"> <li>• Negotiates RFC 2833 payload type using local and remote SDPs.</li> <li>• Sends DTMF packets using RFC 2833 PT according to the received SDP.</li> <li>• Expects to receive RFC 2833 packets with the same PT as configured by the "RFC2833PayloadType" parameter.</li> </ul> <b>Note 3:</b> If the remote party doesn't include the RFC 2833 DTMF relay payload type in the SDP, the gateway uses the same PT for send and for receive. <b>Note 4:</b> If TxDTMFOption is set to 0, the RFC 2833 payload type is set according to the parameter 'RFC2833PayloadType' for both transmit and receive.
<b>IsDTMFUsed</b> [Use Out-of-Band DTMF]	Use out-of-band signaling to relay DTMF digits. 0 = Disable, DTMF digits are sent according to DTMFTransportType parameter. (default) 1 = Enable sending DTMF digits within INFO or NOTIFY messages. When out-of-band DTMF transfer is used DTMFTransportType is automatically set to 0.

**Table 6-3: SIP Configuration Parameters (continues on pages 100 to 110)**

<i>ini</i> File Field Name Web Parameter Name *	Valid Range and Description
<b>OutOfBandDTMFFormat</b> [Out-of-Band DTMF Format]	The exact method to send out-of-band DTMF digits 1 = INFO format (Nortel) 2 = INFO format (Cisco) - (default) 3 = NOTIFY format <draft-mahy-sipping-signaled-digits-01.txt>  <b>Note 1:</b> To use out-of-band DTMF, set "IsDTMFUsed=1". <b>Note 2:</b> When using out-of-band DTMF, the "DTMFTransportType" parameter is automatically set to 0, to erase the DTMF digits from RTP path.
<b>DisableAutoDTMFmute</b>	Enables / disables the automatic mute of DTMF digits when out-of-band DTMF transmission is used. 0 = Auto mute is used (default). 1 = No automatic mute of in-band DTMF.  When 'DisableAutoDTMFmute=1', the DTMF transport type is set according to the parameter 'DTMFTransportType' and the DTMF digits aren't muted if out-of-band DTMF mode is selected ('IsDTMFUsed=1'). This enables the sending of DTMF digits in-band (transparent of RFC 2833) in addition to out-of-band DTMF messages. <b>Note:</b> Usually this mode is not recommended.
<b>MaxActiveCalls</b> [Max Number Of Active Calls]	Defines the maximum number of calls that the gateway can have active at the same time. If the maximum number of calls is reached, new calls are not established. The default value is max available channels (no restriction on the maximum number of calls). The valid range is 1 to 240.
<b>MaxCallDuration</b> [Max Call Duration]	Defines the maximum call duration in minutes. If this time expires, both sides of the call are released (IP and Tel). The valid range is 0 to 120. The default is 0 (no limitation).
<b>EnableBusyOut</b> [Enable Busy Out]	0 = Not used (default) 1 = Enable busy out If Proxy is not responding (according to the Proxy keep alive mechanism) or if there is a failure in the network, and if fallback isn't enabled (IsFallbackUsed=0), all E1/T1 trunks are automatically put out of service by sending a remote alarm (AIS) or Service Out message for T1 PRI trunks that support these messages (NI-2, 4/5-ESS, DMS-100 and Meridian). Note that behavior varies between different protocol types.
<b>EnableDigitDelivery2IP</b> [Enable Digit Delivery to IP]	0 = Disabled (default). 1 = Enable digit delivery to IP. The digit delivery feature enables sending of DTMF digits to the destination IP address after the Tel→IP call was answered. To enable this feature, modify the called number to include at least one 'p' character. The gateway uses the digits before the 'p' character in the initial Invite message. After the call was answered the gateway waits for the required time (# of 'p' * 1.5 seconds) and then sends the rest of the DTMF digits using the method chosen (in-band, out-of-band). <b>Note:</b> The called number can include several 'p' characters (1.5 seconds pause). For example, the called number can be as follows: pp699, p9p300.
<b>EnableDigitDelivery</b> [Enable Digit Delivery to Tel]	The digit delivery feature enables sending of DTMF digits to the Gateway's B-channel after the call is answered. 0 = Disabled (default). 1 = Enable Digit Delivery feature for Mediant 2000 (two stage dialing). <b>Note:</b> For incoming IP→Tel calls, if the called number includes the characters 'w' or 'p', the Mediant 2000 Gateway places a call with the first part of the called number, and plays DTMF digits after the call is answered. If the character 'p' (pause) is used, the Mediant 2000 waits for 1.5 seconds before playing the next DTMF digit. If the character 'w' is used, the Mediant 2000 waits for detection of dial tone before it starts playing DTMF digits. The character 'w' can appear once in the called number, and must precede any 'p' character. The 'p' character can appear several times. For example: if the number "1007766p100" is defined as the called number, the Mediant 2000 places a call with 1007766 as the destination number, then, after the call is answered, it waits for 1.5 seconds and plays the rest of the number (100) as DTMF digits. Other examples: 1664wpp102, 66644ppp503, 7774w100pp200.

**Table 6-3: SIP Configuration Parameters (continues on pages 100 to 110)**

<i>ini</i> File Field Name Web Parameter Name *	Valid Range and Description
<b>Profile Parameters</b>	
<b>CoderName_ID</b> [Coder Group Settings]	<p>Coder list for Profiles (up to five coders in each group).                      The CoderName_ID parameter (ID from 1 to 4) provides groups of coders that can be associated with IP or Tel profiles.</p> <p>You can select the following coders:</p> <ul style="list-style-type: none"> <li>g711Alaw64k – G.711 A-law.</li> <li>g711Ulaw64k – G.711 <math>\mu</math>-law.</li> <li>g7231 – G.723.1 6.3 kbps (default).</li> <li>g7231r53 – G.723.1 5.3 kbps.</li> <li>g726 – G.726 ADPCM 32 kbps (Payload Type = 2).</li> <li>g729 – G.729A.</li> <li>NetCoder6_4 – NetCoder 6.4 kbps.</li> <li>NetCoder7_2 – NetCoder 7.2 kbps.</li> <li>NetCoder8 – NetCoder 8.0 kbps.</li> <li>NetCoder8_8 – NetCoder 8.8 kbps.</li> <li>Transparent – Transparent coder.</li> <li>EvrC – EVRC coder.</li> <li>Amr – AMR coder.</li> </ul> <p>The RTP packetization period (ptime, in msec) depends on the selected Coder name, and can have the following values:</p> <ul style="list-style-type: none"> <li>g711 family – 10, 20, 30, 40, 50, 60, 80, 100, 120 (default=20).</li> <li>g729 – 10, 20, 30, 40, 50, 60, 80, 100 (default=20).</li> <li>g723 family – 30, 60, 90, 120 (default = 30).</li> <li>G.726 family – 10, 20, 30, 40, 50, 60, 80, 100, 120 (default=20)</li> <li>NetCoder family – 20, 40, 60, 80, 100, 120 (default=20).</li> <li>EVRC – 20, 40, 60, 80, 100 (default=20).</li> <li>AMR – 20 only.</li> <li>Transparent – 20, 40, 60, 80, 100, 120 (default=20)</li> </ul> <p><b>Note 1:</b> If not specified, the ptime gets a default value.  <b>Note 2:</b> Each coder should appear only once.  <b>Note 3:</b> The ptime specifies the maximum packetization time the Gateway will receive.  <b>Note 4:</b> G.729B is supported if the coder G.729 is selected and 'EnableSilenceCompression' equals 1 or 2.  <b>Note 5:</b> The selected rate of the AMR coder is set according to the parameter 'AMRSendRate'. The selected rate of the EVRC coder is set according to the parameter 'EVRCRate'.  <b>Note 6:</b> The AMR coder is enabled only if 'DSPVersionTemplateNumber = 1'. When this DSP template is selected, the maximum number of channels is 160 instead of 240 (rounded to full E1/T1 trunk capacity 30/24 channels per trunk).  <b>Note 7:</b> The EVRC coder is enabled only if 'DSPVersionTemplateNumber = 2'. When this DSP template is selected, the maximum number of channels is 120 instead of 240 (rounded to full E1/T1 trunk capacity 30/24 channels per trunk). Note that the value of the parameter 'EnableRFC2658Interleaving' must be identical on both sides of the call.</p> <p><b>ini file note 1:</b> This parameter (CoderName_ID) can appear up to 20 times (five coders in four coder groups).  <b>ini file note 2:</b> The coder name is case-sensitive.  <b>ini file note 3:</b> Enter in the format: CoderName,ptime.</p> <p>For example, the following three coders belong to coder group with ID=1:                      CoderName_1 = g711Alaw64k,20                      CoderName_1 = g711Ulaw64k,40                      CoderName_1 = g7231,90</p>

**Table 6-3: SIP Configuration Parameters (continues on pages 100 to 110)**

<i>ini</i> File Field Name Web Parameter Name *	Valid Range and Description
<b>IPProfile_ID</b> [IP Profile Settings]	<p>IPProfile_&lt;Profile ID&gt; = &lt;Profile Name&gt;,&lt;Preference&gt;,&lt;Coder Group ID&gt;,&lt;IsFaxUsed *&gt;,&lt;DJBufMinDelay *&gt;,&lt;DJBufOptFactor *&gt;,&lt;IPDiffServ *&gt;,&lt;ControllIPDiffServ *&gt;,&lt;EnableSilenceCompression&gt;,&lt;RTPRedundancyDepth&gt;</p> <p>Preference = (1-10) The preference option is used to determine the priority of the Profile. If both IP and Tel profiles apply to the same call, the coders and other common parameters of the preferred Profile are applied to that call. If the Preference of the Tel and IP Profiles is identical, the Tel Profile parameters are applied.</p> <p>For example: IPProfile_1 = name1,2,1,0,10,13,15,44,1,1 IPProfile_2 = name2,\$,\$,\$,\$,\$,\$,\$,\$,\$,\$,1</p> <p>\$\$ = Not configured, the default value of the parameter is used. (* ) = Common parameter used in both IP and Tel profiles.</p> <p><b>Note 1:</b> The IP ProfileID can be used in the Tel2IP and IP2Tel routing tables (Prefix and PSTNPrefix parameters). <b>Note 2:</b> 'Profile Name' assigned to a ProfileID, enabling User's to identify it intuitively and easily. <b>Note 3:</b> This parameter can appear up to 4 times.</p>
<b>TelProfile_ID</b> [Tel Profile Settings]	<p>TelProfile_&lt;Profile ID&gt; = &lt;Profile Name&gt;,&lt;Preference&gt;,&lt;Coder Group ID&gt;,&lt;IsFaxUsed *&gt;,&lt;DJBufMinDelay *&gt;,&lt;DJBufOptFactor *&gt;,&lt;IPDiffServ *&gt;,&lt;ControllIPDiffServ*&gt;,&lt;DtmfVolume&gt;,&lt;InputGain&gt;,&lt;VoiceVolume&gt;,&lt;EnableDigitDelivery&gt;,&lt;ECE&gt;</p> <p>Preference = (1-10) The preference option is used to determine the priority of the Profile. If both IP and Tel profiles apply to the same call, the coders and other common parameters of the preferred Profile are applied to that call. If the Preference of the Tel and IP Profiles is identical, the Tel Profile parameters are applied.</p> <p>For examples: TelProfile_1 = FaxProfile,1,2,0,10,5,22,33,2,22,34,1,1 TelProfile_2 = ModemProfile,0,10,13,\$,\$,\$,\$,\$,\$,\$,0,\$\$,0,1</p> <p>\$\$ = Not configured, the default value of the parameter is used. (* ) = Common parameter used in both IP and Tel profiles.</p> <p><b>Note 1:</b> The Tel ProfileID can be used in the Trunk Group table (TrunkGroup_x parameter). <b>Note 2:</b> 'Profile Name' assigned to a ProfileID, enabling User's to identify it intuitively and easily. <b>Note 3:</b> This parameter can appear up to 4 times.</p>

## 6.9 ISDN and CAS Interworking-Related Parameters



**Note:** In Table 6-4, parameters in brackets are the format in the Embedded Web Server \*.

Table 6-4: ISDN and CAS Interworking-Related Parameters (continues on pages 111 to 114)

<i>ini</i> File Field Name Web Parameter Name *	Valid Range and Description
<b>EnableTDMoverIP</b> [Enable TDM Tunneling]	<p>0 = Disabled (default). 1 = TDM Tunneling is enabled.</p> <p>When TDM Tunneling is enabled, the originating Mediant 2000 automatically initiates SIP calls from all enabled B-channels belonging to the E1/T1/J1 spans that are configured with the 'Transparent' protocol. The called number of each call is the internal phone number of the B-channel that the call originates from. The IP to Trunk Group routing table is used to define the destination IP address of the terminating Mediant 2000. The terminating Mediant 2000 gateway automatically answers these calls if its E1/T1 protocol is set to 'Transparent' (ProtocolType = 5).</p>
<b>PlayRBTone2Tel</b> [Play Ringback Tone to Tel]	<p>0 (Don't play) = The ISDN / CAS gateway doesn't play a Ringback Tone (RBT). No PI is sent to the ISDN, unless the parameter 'Progress Indicator to ISDN' is configured differently. 1 (Play) = The CAS gateway plays a local RBT to PSTN after receipt of a 180 ringing response (with or without SDP). <b>Note:</b> Reception of a 183 response doesn't cause the CAS gateway to play an RBT (unless 'SIP183Behavior = 1'). The ISDN gateway functions according to the parameter 'LocalISDNRBSsource':</p> <ul style="list-style-type: none"> <li>• If the ISDN gateway receives a 180 ringing response (with or without SDP) and 'LocalISDNRBSsource = 1', it plays a RBT and sends an Alert with PI = 8 (unless the parameter 'Progress Indicator to ISDN' is configured differently).</li> <li>• If 'LocalISDNRBSsource = 0', the ISDN gateway doesn't play an RBT and an Alert message (without PI) is sent to the ISDN. In this case, the PBX / PSTN should play the RBT to the originating terminal by itself.</li> </ul> <p><b>Note:</b> Reception of a 183 response doesn't cause the ISDN gateway to play an RBT; the gateway issues a Progress message (unless 'SIP183Behavior = 1'). If 'SIP183Behavior = 1', the 183 response is treated the same way as a 180 ringing response.</p> <p>2 = Play according to "early media" (default). If a 180 response is received and the voice channel is already open (due to a previous 183 early media response or due to an SDP in the current 180 response), the ISDN / CAS gateway doesn't play the RBT; PI = 8 is sent in an ISDN Alert message (unless the parameter 'Progress Indicator to ISDN' is configured differently).</p> <p>If a 180 response is received but the "early media" voice channel is not opened, the CAS gateway plays an RBT to the PSTN; the ISDN gateway functions according to the parameter 'LocalISDNRBSsource':</p> <ul style="list-style-type: none"> <li>• If 'LocalISDNRBSsource = 1', the ISDN gateway plays an RBT and sends an ISDN Alert with PI = 8 to the ISDN (unless the parameter 'Progress Indicator to ISDN' is configured differently).</li> <li>• If 'LocalISDNRBSsource = 0', the ISDN gateway doesn't play an RBT. No PI is sent in the ISDN Alert message (unless the parameter 'Progress Indicator to ISDN' is configured differently). In this case, the PBX / PSTN should play an RBT tone to the originating terminal by itself.</li> </ul> <p><b>Note:</b> Reception of a 183 response results in an ISDN Progress message (unless 'SIP183Behavior = 1'). If 'SIP183Behavior = 1' (183 is handled in the same way as a 180+SDP), the gateway sends an Alert message with PI = 8, without playing an RBT.</p>

**Table 6-4: ISDN and CAS Interworking-Related Parameters (continues on pages 111 to 114)**

<b>ini File Field Name Web Parameter Name *</b>	<b>Valid Range and Description</b>
<b>PlayRBTone2IP</b> [Play Ringback Tone to IP]	0 = Ringback tone isn't played (default). 1 = Ringback tone is played (to IP) after SIP 183 session progress response is sent.  If configured to '1' (Play), For IP→Tel calls, if a Progress or an Alert message with PI is sent from the ISDN and 'EnableEarlyMedia = 1', the Mediant 2000 opens a voice channel and sends 183 response. It doesn't play a Ringback tone to IP (assuming that the Ringback tone is played by the ISDN). Otherwise, if a voice channel isn't opened, the Mediant 2000 plays Ringback tone to IP, after receiving an Alert message from the ISDN. It sends 183 response, signaling the originating party to open a voice channel in order to hear the played Ringback tone. <b>Note 1:</b> To enable the gateway to send a 183 response, set 'EnableEarlyMedia' to 1. <b>Note 2:</b> If 'EnableDigitDelivery = 1', the gateway doesn't play a Ringback tone to IP and doesn't send a 183 response.
<b>ProgressIndicator2ISDN</b> [Progress Indicator to ISDN]	0, 1 or 8 -1 = Not configured (default). If set to "0" PI is not sent to ISDN If set to "1" or "8" the PI value is sent to PSTN in Q.931/Proceeding and Alerting messages. If not configured, the PI in ISDN messages is set according to the "Play Ringback to Tel" parameter. Usually if PI = 1 or 8, the PSTN/PBX cuts through the audio channel without playing local Ringback tone, enabling the originating party to hear remote Call Progress Tones or network announcements
<b>ProgressIndicator2IP</b> [Progress Indicator to IP]	-1 = (Not configured) for ISDN spans, the PI that is received in ISDN Proceeding, Progress and Alert messages is used as described in the following options (default). 0 = (No PI) For IP→Tel call, the gateway sends "180 Ringing" SIP response to IP after receiving ISDN Alert, or (for CAS) after placing a call to PBX/PSTN. 8, 1 = For IP→Tel call, if 'EnableEarlyMedia=1', the gateway sends "183 session in progress" message + SDP, after a call is placed to PBX/PSTN over the trunk. This is used to cut through voice path, before remote party answers the call, enabling the originating party to listen to network Call Progress Tones (such as Ringback tone or other network announcements).
<b>PIForDisconnectMsg</b> [Set PI in Rx Disconnect Message]	Defines the gateway's behavior when a Disconnect message is received from the ISDN before a Connect message was received. -1 (Not configured) = Sends a 183 message according to the received PI in the ISDN Disconnect message. If PI = 1 or 8, the gateway sends a 183 response, enabling the PSTN to play a voice announcement to the IP side. If there isn't a PI in the Disconnect message, the call is released (default). 0 = Do not send a 183 message to IP. The call is released. 1, 8 = Sends 183 message to IP.
<b>ConnectOnProgressInd</b>	0 = Connect message isn't sent after 183 Session Progress is received (default). 1 = Connect message is sent after 183 Session Progress is received. This feature enables the play of announcements from IP to PSTN without the need to answer the Tel→IP call. It can be used with PSTN networks that don't support the opening of a TDM channel before an ISDN Connect message is received.
<b>SIP183Behavior</b> [183 Message Behavior]	Defines the ISDN message that is sent when 183 Session Progress message is received for IP→Tel calls. 0 = Progress message (default). 1 = Alert message. When set to 1, the gateway sends an Alert message (after the receipt of a 183 response) instead of an ISDN Progress message.
<b>LocalISDNRBSSource</b> [Local ISDN Ringback Tone Source]	Determines whether Ringback tone is played to the ISDN by the PBX / PSTN or by the gateway. 0 = PBX / PSTN (default). 1 = Gateway. This parameter is applicable to ISDN protocols. It is used simultaneously with the parameter 'PlayRBTone2Tel'.

Table 6-4: ISDN and CAS Interworking-Related Parameters (continues on pages 111 to 114)

<i>ini</i> File Field Name Web Parameter Name *	Valid Range and Description
<b>PSTNAlertTimeout</b> [PSTN Alert Timeout]	Alert Timeout in seconds (ISDN T301 timer) for outgoing calls to PSTN. The default is 180 seconds. The range is 0 to 240. <b>Note:</b> The PSTN stack T301 timer can be overridden by a lower value, but it can't be increased.
<b>ISDNTransferCapability</b> [ISDN Transfer Capabilities]	Defines the IP→ISDN Transfer Capability of the Bearer Capability IE in ISDN Setup messages. 0 = Audio 3.1 (default). 1 = Speech. 2 = Data. <b>Note:</b> If this parameter isn't configured or equals to '-1', Audio 3.1 capability is used.
<b>ScreeningInd2IP</b> [Send Screening Indicator to IP]	The parameter can overwrite the calling number screening indication for ISDN Tel→IP calls. -1 = not configured (interworking from ISDN to IP) or set to 0 for CAS. 0 = user provided, not screened. 1 = user provided, verified and passed. 2 = user provided, verified and failed. 3 = network provided. <b>Note:</b> Applicable only if Remote Party ID (RPID) header is enabled.
<b>SupportRedirectInFacility</b>	0 = Not Supported (default). 1 = Supports partial retrieval of Redirect Number (number only) from a Facility IE in ISDN Setup messages. Applicable to Redirect number according to ECMA-173 Call Diversion Supplementary Services. <b>Note:</b> To enable this feature, 'ISDNDuplicateQ931BuffMode' must be set to 1.
<b>EnableCIC</b>	0 = Do not relay the Carrier Identification Code (CIC) to ISDN (default). 1 = CIC is relayed to ISDN in Transit Network Selection IE. If enabled, the CIC code (received in an Invite Request-URI) is included in a TNS IE in ISDN Setup message. For example: INVITE sip:555666;cic=2345@100.2.3.4 sip/2.0. <b>Note:</b> Currently this feature is supported only in SIP→ISDN direction.
<b>EnableAOC</b>	0 = Not used (default). 1 = ISDN Advice of Charge (AOC) messages are interworked to SIP.  The gateway supports reception of ISDN (Euro ISDN) AOC messages. AOC messages can be received during a call (Facility messages) or at the end of a call (Disconnect or Release messages). The gateway converts the AOC messages into SIP Info (during a call) and Bye (end of a call) messages using a proprietary AOC SIP header. The gateway supports both Currency and Pulse AOC messages.
<b>TimeForReorderTone</b>	Busy or Reorder Tone duration the CAS gateway plays before releasing the line. The valid range is 0 to 15. The default value is 10 seconds. Applicable also to ISDN if 'PlayBusyTone2ISDN = 2'. Selection of Busy or Reorder tone is done according to release cause received from IP.
<b>DisconnectOnBusyTone</b> [Disconnect Call on Detection of Busy Tone]	0 = Do not disconnect call on detection of busy tone 1 = Disconnect call on detection of busy tone (default). This parameter is applicable to CAS & ISDN protocols.
<b>PlayBusyTone2ISDN</b> [Play Busy Tone to Tel]	This parameter enables the Mediant 2000 ISDN gateway to play a Busy or a Reorder tone to the PSTN after a call is released. 0 = Immediately sends an ISDN Disconnect message (default). 1 = Sends an ISDN Disconnect message with PI=8 and plays a Busy or a Reorder tone to the PSTN (depending on the release cause). 2 = Delays the sending of an ISDN Disconnect message for 'TimeForReorderTone' seconds and plays a Busy or a Reorder tone to the PSTN. Applicable only if the call is released from the IP before it reaches the Connect state. Otherwise, the Disconnect message is sent immediately and no tones are played.
<b>TrunkTransferMode_X</b>	0 = Not supported (default). 1 = Supports CAS NFA DMS-100 transfer. When a SIP Refer message is received, the gateway performs a Blind Transfer by executing a CAS Wink and dialing the Refer-to number to the Switch and then releasing the call. <b>Note:</b> A specific NFA CAS table is required.

**Table 6-4: ISDN and CAS Interworking-Related Parameters (continues on pages 111 to 114)**

<i>ini</i> File Field Name Web Parameter Name *	Valid Range and Description
<b>CASTransportType</b> [CAS Transport Type]	0 = Disable CAS relay (default). 1 = Enable CAS relay mode using RFC 2833. The CAS relay mode can be used with the TDM tunneling feature to enable tunneling over IP for both voice and CAS signaling bearers.
<b>XChannelHeader</b>	0 = x-channel header is not used (default). 1 = x-channel header is generated, with trunk/B-channel information.  The header provides information on the E1/T1 physical trunk/B-channel on which the call is received or placed. For example "x-channel: DS/DS1-5/22". This header is generated by the Mediant 2000 and is sent in the following messages: INVITE and 183/180/200OK responses.
<b>AddIEinSetup</b> [Add IE in SETUP]	This parameter enables to add an optional Information Element data (in hex format) to ISDN SETUP message. For example: to add the following IE: "0x20,0x02,0x00,0xe1", define: "AddIEinSetup = 200200e1".  <b>Note:</b> This IE is sent from the Trunk Group IDs defined by the parameter 'SendIEonTG'.
<b>SendIEonTG</b> [Trunk Groups to Send IE]	A list of Trunk Group IDs (up to 50 characters) from where the optional ISDN IE, defined by the parameter 'AddIEinSetup', is sent. For example: "SendIEonTG = 1,2,4,10,12,6".
<b>ISDNDMSTimerT310</b>	Overrides the T310 timer for the DMS-100 ISDN variant. T310 defines the timeout between the reception of Proceeding message and the reception of Alert / Connect message. The valid range is 10 to 30. The default value is 10 (seconds). <b>Note:</b> Applicable only to Nortel DMS and Nortel MERIDIAN PRI variants (ProtocolType = 14 and 35).
<b>ISDNJapanNTTTimerT3JA</b>	T3_JA timer (in seconds). This parameter overrides the internal PSTN T301 timeout on the Users Side (TE side). If an outgoing call from the Mediant 2000 to ISDN is not answered during this timeout, the call is released. The valid range is 10 to 240. The default value is 50. Applicable only to Japan NTT PRI variant (ProtocolType = 16). <b>Note:</b> This timer is also affected by the parameter 'PSTNAlertTimeout'.

## 6.10 Number Manipulation and Routing Parameters



**Note:** In Table 6-5, parameters in brackets are the format in the Embedded Web Server \*.

**Table 6-5: Number Manipulation and Routing Parameters (continues on pages 115 to 121)**

<i>ini</i> File Field Name Web Parameter Name *	Valid Range and Description
<b>TrunkGroup_x</b> [Trunk Group Table]	TrunkGroup_x = T/a-b,c,d  x = Trunk group ID (1 to 99). T = Physical trunk number (0 to 7). a = Starting B-channel (from 1). b = Ending B-channel (up to 31). c = Phone number associated with the first channel (optional). d = Optional Tel Profile ID (1 to 5).  For example: TrunkGroup_1 = 0/1-31,1000 (for E1 span). TrunkGroup_1 = 1/1-31,\$\$,1. TrunkGroup_2 = 2/1-24,3000 (for T1 span).  Trunk group is the recommended method to configure the gateway's B-channels. The parameter 'ChannelList' (although still supported) mustn't be used simultaneously with Trunk Groups. <b>Note:</b> An optional Tel Profile ID (1 to 5) can be applied to each group of B-channels.
<b>ChannelList</b>  <b>Note:</b> It is recommended to use <b>TrunkGroup_x</b> parameter instead.	List of phone numbers, used to define the enabled B-channels for gateway operation, 'a, b, c,d' a = first channel b = number of channels starting from 'a' c = the phone number of the first channel d = Tel Profile ID assigned to the group of channels. example: ChannelList = '0,30,1001' Defines phone numbers 1001 to 1030 for 30 gateway channels. The <i>ini</i> file can include up to ten 'ChannelList = ' entries Usually single ChannelList parameter is enough to define the complete 8 trunk gateway: ChannelList = '0,240,1000'; For eight E1 spans ChannelList = '0,192,1000'; For eight T1 CAS spans Phone numbers can be defined individually per E1 or T1 span: For E1 spans (CAS or ISDN): 0 to 29 for first span, 30 to 59 for second span, 60 to 89 for 3 <sup>rd</sup> span, 90 to 119 for 4 <sup>th</sup> span. For T1 ISDN spans: 0 to 22 for first span, 23 to 45 for second span, 46 to 68 for 3 <sup>rd</sup> span, and 69 to 91 for 4 <sup>th</sup> span. For T1 CAS signaling: 0 to 23 for first span, 24 to 47 for second span, 48 to 71 for 3 <sup>rd</sup> span, and 72 to 95 for 4 <sup>th</sup> span.  It is suggested to use Trunk Groups in Mediant 2000 gateway to define enabled B-channels, instead of ChannelList parameter.
<b>DefaultNumber</b> [Default Destination Number]	Defines the telephone number that the gateway uses if the parameters 'TrunkGroup_x' or 'ChannelList' don't include a phone number. The parameter is used as a starting number for the list of B-channels comprising all trunk groups in the gateway.

**Table 6-5: Number Manipulation and Routing Parameters (continues on pages 115 to 121)**

<i>ini</i> File Field Name Web Parameter Name	Valid Range and Description
<b>ChannelSelectMode</b> [Channel Select Mode]	Defines common rule of port allocation for IP to TEL calls. <ul style="list-style-type: none"> <li>• 0 = By phone number - Select the gateway port according to the called number (refer to the note below).</li> <li>• 1 = Cyclic Ascending - Select the next available channel in an ascending cycle order. Always select the next higher channel number in the Trunk Group. When the gateway reaches the highest channel number in the Trunk Group, it selects the lowest channel number in the Trunk Group and then starts ascending again (default).</li> <li>• 2 = Ascending - Select the lowest available channel. Always start at the lowest channel number in the Trunk Group and if that channel is not available, select the next higher channel.</li> <li>• 3 = Cyclic Descending - Select the next available channel in descending cycle order. Always select the next lower channel number in the Trunk Group. When the gateway reaches the lowest channel number in the Trunk Group, it selects the highest channel number in the Trunk Group and then start descending again.</li> <li>• 4 = Descending - Select the highest available channel. Always start at the highest channel number in the Trunk Group and if that channel is not available, select the next lower channel.</li> <li>• 5 = Number + Cyclic Ascending – First select the gateway port according to the called number (refer to the note below). If the called number isn't found, then select the next available channel in ascending cyclic order. Note that if the called number is found, but the port associated with this number is busy, the call is released.</li> </ul> <p><b>Note:</b> The internal numbers of the gateway's B-channels are defined by the 'TrunkGroup_x' parameter (under 'Phone Number').</p>
<b>TrunkGroupSettings</b> [Trunk Group Settings]	Defines rules for port allocation for specific Trunk Groups, if such rule doesn't exist, the global rule defined by ChannelSelectMode applies. a, b a = Trunk Group ID number b = Channel select mode for that Trunk Group. Available values are identical to those defined by the ChannelSelectMode parameter.
<b>AddTrunkGroupAsPrefix</b> [Add Trunk Group ID as Prefix]	0 = not used 1 = For Tel→IP incoming call, Trunk Group ID is added as prefix to destination phone number. Applicable only if trunk group ID are configured. Can be used to define various routing rules.
<b>AddPortAsPrefix</b> [Add Trunk ID as Prefix]	0 = Don't add (default) 1 = Add trunk ID number (single digit in the range 1 to 8) as a prefix to the called phone number for Tel→IP incoming calls. This option can be used to define various routing rules.
<b>ReplaceEmptyDstWithPortNumber</b> [Replace Empty Destination with Port Number]	0 = Disabled (default). 1 = Enabled, Internal channel number is used as a destination number if called number is missing. <b>Note:</b> Applicable only to Tel→IP calls, if called number is missing.
<b>CopyDestOnEmptySource</b>	0 = Leave Source Number empty (default). 1 = If the Source Number of an incoming Tel to IP call is empty, the Destination Number is copied to the Source Number.
<b>AddNPIandTON2CallingNumber</b>	0 = Do not change the Calling Number (default). 1 = Add NPI and TON to the Calling Number of incoming (Tel to IP) ISDN call. For example: After receiving a Calling Number = 555, NPI = 1 and TON = 3, the modified number is going to be 13555. This number can later be used for manipulation and routing purposes.
<b>AddNPIandTON2CalledNumber</b>	0 = Do not change the Called Number (default). 1 = Add NPI and TON to the Called Number of incoming (Tel to IP) ISDN call. For example: After receiving a Called Number = 555, NPI=1 and TON = 3, the modified number is going to be 13555. This number can later be used for manipulation and routing purposes.

Table 6-5: Number Manipulation and Routing Parameters (continues on pages 115 to 121)

<b>ini File Field Name</b> <b>Web Parameter Name</b>	<b>Valid Range and Description</b>
<b>UseSourceNumberAsDisplay Name</b> [Use Source Number as Display Name]	0 = Interworks the Tel calling name to SIP Display Name (default). 1 = Set Display Name to Calling Number if not configured.  Applicable to Tel→IP calls. If enabled and if the incoming Tel to IP call doesn't contain the calling party name, the calling number is used instead. All CAS protocols don't provide the calling party name. Therefore, in CAS, if this parameter is enabled, the Display Name is identical to the calling number.
<b>AlwaysUseRouteTable</b> [Use Routing Table for Host Names and Profiles]	Use the internal Tel to IP routing table to obtain the URL Host name and (optionally) an IP profile (per call), even if Proxy server is used. 0 = Don't use (default) 1 = Use <b>Note:</b> This Domain name is used, instead of Proxy name or Proxy IP address, in the INVITE SIP URL.
<b>Prefix</b> [Tel to IP Routing Table]	Mapping phone number to IP address, using phone number prefix Selection of IP address (for Tel To IP calls) is according to destination and source prefixes. Prefix = <Destination Phone Prefix>, <IP Address>, <Src Phone Prefix>, <IP Profile ID>  For example: Prefix = 20,10.2.10.2,202,1 Prefix = 10[340-451]xxx#,10.2.10.6,*,1 Prefix = *,gateway.domain.com,*  <b>Note 1:</b> An optional IP ProfileID (1 to 5) can be applied to each routing rule. <b>Note 2:</b> <destination / source phone prefix> can be single number or a range of numbers. <b>Note 3:</b> This parameter can appear up to 50 times. <b>Note 4:</b> Parameters can be skipped by using the sign "\$\$", for example: Prefix = \$\$,10.2.10.2,202,1 For available notations, refer to Section 5.8.3.1 on page 47. For detailed information on this feature, refer to Section 5.8.4.1 on page 49.
<b>PSTNPrefix</b> [IP to Trunk Group Routing Table]	PSTNPrefix = a,b,c,d,e  a = Destination Number Prefix b = Trunk group ID (1 to 99) c = Source Number Prefix d = Source IP address (obtained from the Contact header in the Invite message) e = IP Profile ID (1 to 5)  Selection of trunk groups (for IP to Tel calls) is according to destination number, source number and source IP address.  <b>Note 1:</b> To support the 'in call alternative routing' feature, Users can use two entries that support the same call, but assigned it with a different trunk groups. The second entree functions as an alternative selection if the first rule fails as a result of one of the release reasons listed in the AltRouteCauseIP2Tel table. <b>Note 2:</b> An optional IP ProfileID (1 to 5) can be applied to each routing rule. <b>Note 3:</b> The Source IP Address can include the "x" wildcard to represent single digits. For example: 10.8.8.xx represents all IP addresses between 10.8.8.10 to 10.8.8.99.
<b>RemovePrefix</b> [IP to Tel Remove Routing Table Prefix]	0 = Don't remove prefix (default) 1 = Remove PSTN prefix (defined in the routing table) from a telephone number of an incoming IP call, before forwarding it to PSTN, Applicable only if number manipulation is performed after call routing for IP→Tel calls (RouteModeIP2Tel = 0).
<b>RouteModeIP2Tel</b> [IP to Tel routing Mode]	0 = Route calls before number manipulation (default) 1 = Route calls after number manipulation Defines order between routing calls to Trunk group and manipulation of destination number

**Table 6-5: Number Manipulation and Routing Parameters (continues on pages 115 to 121)**

<i>ini</i> File Field Name Web Parameter Name *	Valid Range and Description
<b>RouteModeTel2IP</b> [Tel to IP routing Mode]	0 = Route calls before number manipulation (default) 1 = Route calls after number manipulation Defines order between routing incoming calls to IP, using routing table, and manipulation of destination number Not applicable if Outbound Proxy is used.
<b>SwapRedirectNumber</b> [Swap Redirect and Called Numbers]	0 = Don't change numbers (default) 1 = Incoming ISDN call that includes redirect number (sometimes referred as "original called number") uses this number instead of the called number.
<b>AddTON2RPI</b> [Add Number Plan and Type to Remote Party ID Header]	0 = TON/PLAN parameters aren't included in the RPID header. 1 = TON/PLAN parameters are included in the RPID header (default). If RPID header is enabled (EnableRPIHeader = 1) and 'AddTON2RPI=1', it is possible to configure the calling and called number type and number plan using the Number Manipulation tables for Tel→IP calls.
<b>NumberMapTel2IP</b> [Destination Phone Number Manipulation Table for Tel→IP calls]	Manipulates the destination number for Tel to IP calls. NumberMapTel2IP = a,b,c,d,e,f,g  a = Destination number prefix b = Number of stripped digits from the left, or (if brackets are used) from the right. A combination of both options is allowed. c = String to add as prefix, or (if brackets are used) as suffix. A combination of both options is allowed. d = Number of remaining digits from the right e = Number Plan used in RPID header f = Number Type used in RPID header g = Source number prefix  The 'b' to 'f' manipulations rules are applied if the called and calling numbers match the 'a' and 'g' conditions.  The manipulation rules are executed in the following order: 'b', 'd' and 'c'. Parameters can be skipped by using the sign "\$\$", for example: NumberMapTel2IP=01,2,972,\$\$,0,0,\$\$ NumberMapTel2IP=03,(2),667,\$\$,0,0,22
<b>NumberMapIP2Tel</b> [Destination Phone Number Manipulation Table for IP→Tel calls]	Manipulate the destination number for IP to Tel calls. NumberMapIP2Tel = a,b,c,d,e,f,g,h,i  a = Destination number prefix b = Number of stripped digits from the left, or (if brackets are used) from the right. A combination of both options is allowed. c = String to add as prefix, or (if brackets are used) as suffix. A combination of both options is allowed. d = Number of remaining digits from the right e = Q.931 Number Plan f = Q.931 Number Type g = Source number prefix h = Not applicable, set to \$\$ i = Source IP address (obtained from the Contact header in the Invite message)  The 'b' to 'f' manipulation rules are applied if the called and calling numbers match the 'a', 'g' and 'i' conditions.  The manipulation rules are executed in the following order: 'b', 'd' and 'c'. Parameters can be skipped by using the sign "\$\$", for example: NumberMapIP2Tel =01,2,972,\$\$,0,\$\$,034 NumberMapIP2Tel =03,(2),667,\$\$,0,22,\$\$,10.13.77.8 <b>Note:</b> The Source IP address can include the "x" wildcard to represent <u>single</u> digits. For example: 10.8.8.xx represents all the addresses between 10.8.8.10 to 10.8.8.99.

**Table 6-5: Number Manipulation and Routing Parameters (continues on pages 115 to 121)**

<b>ini File Field Name Web Parameter Name *</b>	<b>Valid Range and Description</b>
<p><b>SourceNumberMapTel2IP</b> [Source Phone Number Manipulation Table for Tel→IP calls]</p>	<p>SourceNumberMapTel2IP = a,b,c,d,e,f,g,h</p> <p>a = Source number prefix                      b = Number of stripped digits from the left, or (if in brackets are used) from right. A combination of both options is allowed.                      c = String to add as prefix, or (if in brackets are used) as suffix. A combination of both options is allowed.                      d = Number of remaining digits from the right                      e = Number Plan used in RPID header                      f = Number Type used in RPID header                      g = Destination number prefix                      h = Calling number presentation (0 to allow presentation, 1 to restrict presentation)</p> <p>The 'b' to 'f' and 'h' manipulation rules are applied if the called and calling numbers match the 'a' and 'g' conditions.</p> <p>The manipulation rules are executed in the following order: 'b', 'd' and 'c'. Parameters can be skipped by using the sign "\$\$", for example:                      SourceNumberMapTel2IP=01,2,972,\$\$,0,0,\$\$,1                      SourceNumberMapTel2IP=03,(2),667,\$\$,0,0,22,0</p>
<p><b>SourceNumberMapIP2Tel</b> [Source Phone Number Manipulation Table for IP→Tel calls]</p>	<p>Manipulate the source number for IP to Tel calls.                      SourceNumberMapIP2Tel = a,b,c,d,e,f,g,h</p> <p>a = Source number prefix                      b = Number of stripped digits from the left, or (if brackets are used) from the right. A combination of both options is allowed.                      c = String to add as prefix, or (if brackets are used) as suffix. A combination of both options is allowed.                      d = Number of remaining digits from the right                      e = Q.931 Number Plan                      f = Q.931 Number Type                      g = Destination number prefix                      h = Calling number presentation (0 to allow presentation, 1 to restrict presentation)</p> <p>The 'b' to 'f' and 'h' manipulation rules are applied if the called and calling numbers match the 'a' and 'g' conditions.</p> <p>The manipulation rules are executed in the following order: 'b', 'd' and 'c'. Parameters can be skipped by using the sign "\$\$", for example:                      SourceNumberMapIP2Tel =01,2,972,\$\$,0,\$\$,034,1                      SourceNumberMapIP2Tel =03,(2),667,\$\$,0,22</p>

**Table 6-5: Number Manipulation and Routing Parameters (continues on pages 115 to 121)**

<i>ini</i> File Field Name Web Parameter Name *	Valid Range and Description
<p>For ETSI ISDN variant, the following Number Plan and Type combinations (Plan/Type) are supported in the Destination and Source Manipulation tables:</p> <p>0,0 = Unknown, Unknown                      9,0 = Private, Unknown                      9,1 = Private, Level 2 Regional                      9,2 = Private, Level 1 Regional                      9,3 = Private, PISN Specific                      9,4 = Private, Level 0 Regional (local)                      1,0 = Public(ISDN/E.164), Unknown                      1,1 = Public(ISDN/E.164), International                      1,2 = Public(ISDN/E.164), National                      1,3 = Public(ISDN/E.164), Network Specific                      1,4 = Public(ISDN/E.164), Subscriber                      1,6 = Public(ISDN/E.164), Abbreviated</p> <p>For NI-2 and DMS-100 ISDN variants the valid combinations of TON and NPI for calling and called numbers are (Plan/Type):</p> <p>0/0 - Unknown/Unknown                      1/1 - International number in ISDN/Telephony numbering plan                      1/2 - National number in ISDN/Telephony numbering plan                      1/4 - Subscriber (local) number in ISDN/Telephony numbering plan                      9/4 - Subscriber (local) number in Private numbering plan</p>	
<b>SecureCallsFromIP</b> [IP Security]	<p>0 = Gateway accepts all SIP calls (default).                      1 = Gateway accepts SIP calls only from IP addresses defined in the Tel to IP routing table. The gateway rejects all calls from unknown IP addresses.</p> <p>For detailed information on the Tel to IP Routing table, refer to Section 5.8.4.1 on page 49.</p> <p><b>Note:</b> Specifying the IP address of a Proxy server in the Tel to IP Routing table enables the gateway to only accept calls originating in the Proxy server and rejects all other calls.</p>
<b>AltRouteCauseTel2IP</b> [Reasons for Alternative Routing Table]	<p>Table of call failure reason values received from the IP side. If a call is released as a result of one of these reasons, the gateway tries to find an alternative route to that call in the 'Tel to IP Routing' table.</p> <p>For example:                      AltRouteCauseTel2IP = 486 (Busy here).                      AltRouteCauseTel2IP = 480 (Temporarily unavailable).                      AltRouteCauseTel2IP = 408 (No response).</p> <p><b>Note 1:</b> The 408 reason can be used to specify that there was no response from the remote party to the INVITE request.  <b>Note 2:</b> This parameter can appear up to 5 times.</p>
<b>AltRouteCauseIP2Tel</b> [Reasons for Alternative Routing Table]	<p>Table of call failure reason values received from the pstn side (in Q.931 presentation). If a call is released as a result of one of these reasons, the gateway tries to find an alternative trunk group to that call in the 'IP to Trunk Group Routing' table.</p> <p>For example:                      AltRouteCauseIP2Tel = 3 (No route to destination).                      AltRouteCauseIP2Tel = 1 (Unallocated number).                      AltRouteCauseIP2Tel = 17 (Busy here).</p> <p><b>Note 1:</b> This parameter can appear up to 5 times.  <b>Note 2:</b> If the Mediant 2000 fails to establish a call to the PSTN because it has no available channels in a specific trunk group (e.g., all of the trunk group's channels are occupied, or the trunk group's spans are disconnected or out of sync), it uses the internal release cause '3' (no route to destination). This cause can be used in the 'AltRouteCauseIP2Tel' table to define routing to an alternative trunk group.</p>

Table 6-5: Number Manipulation and Routing Parameters (continues on pages 115 to 121)

<b>ini File Field Name Web Parameter Name</b>	<b>Valid Range and Description</b>
<b>FilterCalls2IP</b> [Filter Calls To IP]	0 = Disabled (default) 1 = Enabled  If the filter calls to IP feature is enabled, then when a Proxy is used, the gateway first checks the Tel→IP routing table before making a call through the Proxy. If the number is not allowed (number isn't listed or a Call Restriction routing rule, IP=0.0.0.0, is applied), the call is released.
<b>Alternative Routing Parameters</b>	
<b>AltRoutingTel2IPEnable</b> [Enable Alt Routing Tel to IP]	Operation modes of the Alternative Routing mechanism: 0 = Disabled (default). 1 = Enabled. 2 = Enabled for status only, not for routing decisions.
<b>AltRoutingTel2IPMode</b> [Alt Routing Tel to IP Mode]	0 (None) = Alternative routing is not used. 1 (Conn) = Alternative routing is performed if ping to initial destination failed. 2 (QoS) = Alternative routing is performed if poor quality of service was detected. 3 (All) = Alternative routing is performed if, either ping to initial destination failed, or poor quality of service was detected, or DNS host name is not resolved (default).  <b>Note:</b> QoS is quantified according to delay and packet loss, calculated according to previous calls. Qos statistics are reset if no new data is received for two minutes. For information on the Alternative Routing feature, refer to Section 8.6 on page 146.
<b>IPConnQoSMaxAllowedPL</b> [Max Allowed Packet Loss for Alt Routing]	Packet loss percentage at which the IP connection is considered a failure. The range is 1% to 20%. The default value is 20%.
<b>IPConnQoSMaxAllowedDelay</b> [Max Allowed Delay for Alt Routing]	Transmission delay (in msec) at which the IP connection is considered a failure. The range is 100 to 1000. The default value is 250 msec.

## 6.11 E1/T1 Configuration Parameters



**Note:** In Table 6-6, parameters in brackets are the format in the Embedded Web Server .

**Table 6-6: E1/T1/J1 Configuration Parameters (continues on pages 122 to 127)**

<i>ini</i> File Field Name Web Parameter Name *	Valid Range and Description
<b>PCMLawSelect</b> [PCM Law Select]	1 = A-law 3 = $\mu$ -Law Usually A-Law is used for E1 spans and $\mu$ -Law for T1 and J1 spans.
<b>FramingMethod</b> [Framing Method]	Selects the framing method to be used for E1/T1 spans. <b>For E1</b> 0 = Multiframe with CRC4 (default, automatic mode, if CRC is identified in the Rx, CRC is sent in Tx, otherwise no CRC). a = Double frame c = Multiframe with CRC4 <b>For T1</b> 0 or D = Extended super frame with CRC6 (default) 1 or B = Super frame D4, F12 (12-Frame multiframe) A = F4 (4-Frame multiframe) C = Extended super frame without CRC6 F = J1 - Japan (ESF with CRC6 and JT)
<b>FramingMethod_x</b> [Framing Method]	Same as the description for parameter 'FramingMethod' for a specific trunk ID (x = 0 to 7).
<b>ProtocolType</b> [Protocol Type]	Sets the PSTN protocol to be used for this trunk. E1_EURO_ISDN = 1, T1_CAS = 2, T1_RAW_CAS = 3, T1_TRANSPARENT = 4, E1_TRANSPARENT_31 = 5, E1_TRANSPARENT_30 = 6, E1_MFCR2 = 7, E1_CAS_R2 = 8, E1_RAW_CAS = 9, T1_NI2_ISDN = 10, T1_4ESS_ISDN = 11, T1_5ESS_9_ISDN = 12, T1_5ESS_10_ISDN = 13, T1_DMS100_ISDN = 14, J1_TRANSPARENT = 15 T1_NTT_ISDN = 16 /* Japan - Nippon Telegraph E1_AUSTEL_ISDN = 17 /* Australian Telecom T1_HKT_ISDN = 18 /* Hong Kong - HKT E1_KOR_ISDN = 19 /* Korean operator T1_HKT_ISDN = 20 /* Hong Kong - HKT over T1 E1_QSIG = 21 /*Basic call only T1_QSIG = 23 /*Basic call only T1_DMS100_Meridian = 35  <b>Note:</b> The Mediant 2000 simultaneously supports different variants of CAS and PRI protocols on different E1/T1 spans (no more than four simultaneous PRI variants).
<b>ProtocolType_x</b> [Protocol Type]	Same as the description for parameter 'ProtocolType' for a specific trunk ID (x = 0 to 7).

Table 6-6: E1/T1/J1 Configuration Parameters (continues on pages 122 to 127)

<b>ini File Field Name</b> <b>Web Parameter Name</b>	<b>Valid Range and Description</b>
<b>TerminationSide</b> [ISDN Termination Side]	Selects the ISDN termination side. Applicable only to ISDN protocols. 0 = ISDN User Termination Side (TE) (default) 1 = ISDN Network Termination Side (NT)  <b>Note:</b> select 'User Side' when the PSTN or PBX side is configured as 'Network side', and vice-versa. If you don't know the Mediant 2000 ISDN termination side, choose 'User Side' and refer to the 'Status & Diagnostics>Channel Status' screen. If the D-channel alarm is indicated, choose 'Network Side'.
<b>TerminationSide_x</b> [ISDN Termination Side]	Same as the description for parameter 'TerminationSide' for a specific trunk ID (x = 0 to 7).
<b>ClockMaster</b> [Clock Master]	0 = Recover clock from the E1/T1 line (default) 1 = The clock is generated by the gateway Refer to <a href="#">Appendix E</a> on page 205 for extended details of how to configure the gateway's clock settings.
<b>ClockMaster_x</b> [Clock Master]	Same as the description for parameter 'ClockMaster' for a specific trunk ID (x = 0 to 7).
<b>TDMBusClockSource</b> [TDM Bus Clock Source]	1 = Generate clock from local source (default). 4 = Recover clock from PSTN line. Refer to <a href="#">Appendix E</a> on page 205 for detailed information on configuring the gateway's clock settings.
<b>TDMBusPSTNAutoClockEnable</b> [TDM Bus PSTN Auto Clock]	0 = Recover the clock from first E1/T1 line (default) 1 = Recover the clock from any connected slave E1/T1 line This parameter is relevant only if "TDMBusClockSource = 4"
<b>TDMBusLocalReference</b> [TDM Bus Local Reference]	0 to 7 (default = 0) Physical Trunk ID from which the gateway recovers its clock. Applicable only if "TDMBusClockSource = 4" and "PSTNAutoClockEnable = 0"
<b>LineCode</b> [Line Code]	0 = use B8ZS line code (for T1 trunks only) default. 1 = use AMI line code. 2 = use HDB3 line code (for E1 trunks only). Use to select B8ZS or AMI for T1 spans, and HDB3 or AMI for E1 spans.
<b>LineCode_x</b> [Line Code]	Same as the description for parameter 'LineCode' for a specific trunk ID (x = 0 to 7).
<b>BchannelNegotiation</b> [B-channel Negotiation]	Determines the ISDN B-Channel negotiation mode. 0 = Preferred 1 = Exclusive (default) Applicable to ISDN protocols.
<b>NFASGroupNumber_x</b> [NFAS Group Number]	0 = Non NFAS trunk (default) 1 to 4 = NFAS group number Indicates the NFAS group number (NFAS member) for the selected trunk. "x" identifies the Trunk ID (0-7). Trunks that belong to the same NFAS group have the same number. With ISDN Non-Facility Associated Signaling you can use single D-channel to control multiple PRI interfaces. Applicable only to T1 ISDN protocols.
<b>DchConfig_x</b> [D-channel Configuration]	0 = Primary Trunk (default) 1 = Backup Trunk 2 = NFAS Trunk D-channel configuration parameter defines primary, backup (optional) and B-channels only trunks. "x" identifies the Trunk ID (0-7). Primary trunk contains D-channel that is used for signaling. Backup trunk contains backup D-channel that is used if the primary D-channel fails. The other NFAS trunks contain only 24 B-channels, without a signaling D-channel. Applicable only to T1 ISDN protocols. Backup trunk is not supported for DMS PRI variants.

**Table 6-6: E1/T1/J1 Configuration Parameters (continues on pages 122 to 127)**

<i>ini</i> File Field Name Web Parameter Name *	Valid Range and Description
<b>ISDNNFASInterfaceID_x</b> [NFAS Interface ID]	<p>Defines a different Interface ID for each T1 trunk. The valid range is 0 to 100. The default interface ID equals to the trunk's ID (0 to 7). 'x' identifies the trunk ID (0-7)</p> <p><b>Note:</b> To set the NFAS interface ID, configure: ISDNIBehavior_x to include '512' feature, per each T1 trunk.</p>
<b>CASTableIndex_x</b> [CAS Table]	<p>Defines CAS protocol for each Trunk ID (x = 0 to 7) from a list of protocols defined by the "CASFileName_Y" parameter. For example: CASFileName_0 = 'E_M_WinkTable.dat' CASFileName_1 = 'E_M_ImmediateTable.dat' CASTableIndex_0 = 0 CASTableIndex_1 = 0 CASTableIndex_2 = 1 CASTableIndex_3 = 1 Trunks 0 and 1 use the E&amp;M Winkstart CAS protocol, while trunks 2 and 3 use the E&amp;M Immediate Start CAS protocol.</p>
<b>CASFileName_0</b> <b>CASFileName_1</b>  <b>CASFileName_7</b>	<p>CAS file name (such as "E_M_WinkTable.dat") defines the CAS protocol. It is possible to define up to 8 different CAS files by repeating the "CASFileName" parameter. Each CAS file can be associated with one or more of the gateway trunks using "CASTableIndex_x" parameter.</p>
<b>CASTablesNum</b>	1 to 8. Indicates how many CAS protocol configurations files are loaded.
<b>IdleABCDPattern</b> [Idle ABCD Pattern]	<p>Range 0x0 to 0xF Default = -1 (default pattern = 0000) ABCD (CAS) Pattern to be applied to CAS signaling bus when the channel is idle. This is only relevant when using PSTN interface with CAS protocols. Set to -1 for default.</p>
<b>IdlePCMPattern</b> [Idle PCM Pattern]	<p>Range 0x00 to 0xFF Default = -1 (default pattern = 0xFF for <math>\mu</math>-Law, 0x55 for A-law) PCM Pattern to be applied to E1/T1 timeslot (B-channel) when the channel is idle.</p>
<b>LineBuildOut.Loss</b> [Line Build Out Loss]	<p>0 = 0 dB (default) 1 = -7.5 dB 2 = -15 dB 3 = -22.5 dB Selects the line build out loss to be used for T1 trunks N/A for E1 trunks.</p>
<b>ISDNRxOverlap</b>	<p>0 = Disabled (default). 1 = Enabled. Any number bigger than one = Number of digits to receive.</p> <p><b>Note 1:</b> If enabled the Mediant 2000 receives ISDN called number that is sent in the "Overlap" mode. <b>Note 2:</b> The INVITE to IP is sent only after the number (including "Sending Complete" Info Element) was fully received (in SETUP and/or subsequent INFO Q.931 messages). For detailed information on ISDN overlap dialing, refer to Section 8.3 on page 140.</p>

Table 6-6: E1/T1/J1 Configuration Parameters (continues on pages 122 to 127)

<b>ini File Field Name Web Parameter Name *</b>	<b>Valid Range and Description</b>
<b>ISDNRxOverlap_x</b> [Enable Receiving of Overlap Dialing]	Enable / disable Rx ISDN overlap per trunk ID (x = 0 to 7). 0 = Disabled (default). 1 = Enabled.  <b>Note 1:</b> If enabled, the Mediant 2000 receives ISDN called number that is sent in the "Overlap" mode. <b>Note 2:</b> The SETUP message to IP is sent only after the number (including the 'Sending Complete' Info Element) was fully received (via SETUP and/or subsequent INFO Q.931 messages). <b>Note3:</b> The 'MaxDigits' parameter can be used to limit the length of the collected number for Mediant 2000 ISDN overlap dialing (if sending complete was not received).
<b>TimeBetweenDigits</b> [Inter Digit Timeout for Overlap Dial]	Defines the time (in seconds) that the gateway waits between digits that are received from the ISDN when Tel→IP overlap dialing is performed. When this inter-digit timeout expires, the gateway uses the collected digits for the called destination number. The range is 1 to 10 seconds. The default value is 4 seconds.
<b>MaxDigits</b> [Max Digits In Phone Num for Overlap Dialing]	Defines the maximum number of collected destination number digits received from the ISDN when Tel→IP overlap dialing is performed. When the number of collected digits reaches the maximum, the gateway uses these digits for the called destination number. The range is 1 to 49. The default value is 30.
<b>DialToneDuration</b>	Duration (in seconds) of the dial tone played to an ISDN terminal. Applicable to overlap dialing when 'ISDNInCallsBehavior = 65536'. The dial tone is played if the ISDN Setup message doesn't include the called number. The valid range is 0 to 60. The default time is 5 seconds.
<b>R2Category</b> [MFC R2 Category]	MFC R2 Calling Party Category (CPC). The parameter provides information on calling party such as National or International call, Operator or Subscriber and Subscriber priority. The parameter range is 1 to 15, defining one of the MFC R2 tones.
<b>RegretTime</b>	Determines the time period (in seconds) the gateway waits for an MFC R2 Resume (Reanswer) signal once a Suspend (Clear back) signal was received from the PBX. If this timer expires, the call is released. The valid range is 0 to 255. The default value is 0. Applicable only for MFC R2 CAS Brazil variant.
<b>HeldTimeout</b>	Determines the time period the gateway can stay on-hold. If a Resume (un-hold Re-Invite) message is received before the timer expires, the call is renewed. If this timer expires, the call is released. -1 = Indefinitely (default). 0 - 2400 = Time to wait in seconds. Currently applicable only to MFC R2 CAS variants.

**Table 6-6: E1/T1/J1 Configuration Parameters (continues on pages 122 to 127)**

<i>ini</i> File Field Name Web Parameter Name	Valid Range and Description
<b>ISDN Flexible Behavior Parameters</b> ISDN protocol is implemented in different Switches / PBXs by different vendors. Several implementations vary a little from the specification. Therefore, to provide a flexible interface that supports these ISDN variants, the ISDN behavior parameters are used.	
<b>ISDNInCallsBehavior</b> [Incoming Calls Behavior]	<p>2048 = Sends Channel ID in the first response to an incoming Q.931 Call Setup message. Otherwise, the Channel ID is sent only if the gateway requires to change the proposed Channel ID (default).</p> <p>8192 = Sends Channel ID in a Q.931 Call Proceeding message.</p> <p>65536 = Includes Progress Indicator (PI=8) in Setup ACK message, if an empty called number is received in an incoming Setup message. Applicable to overlap dialing mode. The parameter also directs the gateway to play a dial tone (for 'DialToneDuration'), until the next called number digits are received.</p> <p>262144 = NI-2 second redirect number – Users can select and use (in Invite messages) the NI-2 second redirect number, if two redirect numbers are received in Q.931 Setup for incoming Tel→IP calls.</p> <p><b>Note:</b> To configure the gateway to support several 'ISDNInCallsBehavior' features, summarize the individual feature values. For example to support both '2048' and '65536' features, set 'ISDNInCallsBehavior = 67584'.</p>
<b>ISDNIBehavior</b> [Q.931 Layer Response Behavior]	<p>1 = Q.931 Status message isn't sent if Q.931 received message contains an unknown/unrecognized IE(s). By default the Status message is sent. This parameter applies only to PRI variants in which sending of Status message is optional.</p> <p>2 = Q.931 Status message isn't sent if an optional IE with invalid content is received. By default the Status message is sent. This parameter applies only to PRI variants in which sending of Status message is optional.</p> <p>4 = Accepts unknown/unrecognized Facility IE. Otherwise, (default) the Q.931 message that contains the unknown Facility IE is rejected. This parameter applies to PRI variants where a complete ASN1 decoding is performed on Facility IE.</p> <p>128 = Connect ACK message is sent in response to received Q.931 Connect. Applicable only to Euro ISDN User side outgoing calls. Otherwise, the Connect ACK is not sent (default).</p> <p>512 = Enables to configure T1 NFAS Interface ID (refer to the parameter 'ISDNNFASInterfaceID_x'). Applicable to 4/5ESS, DMS, NI-2 and HKT variants.</p> <p>2048 = Always set the Channel Identification IE to explicit Interface ID, even if the B-channel is on the same trunk as the D-channel. Applicable to 4/5ESS, DMS and NI-2 variants.</p> <p>65536 = Applicable to ETSI, NI-2 and 5ESS. The calling party number (octet 3a) is always present even when presentation and screening are at their default.</p> <p>131072 = Clears the call on reception of Q.931 Status with incompatible state. Otherwise, (default) no action is taken.</p> <p><b>Note:</b> To configure the gateway to support several 'ISDNIBehavior' features, summarize the individual feature values. For example to support both '512' and '2048' features, set 'ISDNIBehavior = 2560'.</p>

Table 6-6: E1/T1/J1 Configuration Parameters (continues on pages 122 to 127)

<b>ini File Field Name Web Parameter Name</b>	<b>Valid Range and Description</b>
<b>ISDNGeneralCCBehavior</b> [General Call Control Behavior]	<p>16 = The gateway clears down the call if it receives a Notify message specifying 'User-Suspended'. A Notify (User-Suspended) message is used by some networks (e.g., in Italy or Denmark) to indicate that the remote user has cleared the call, especially in the case of a long distance voice call.</p> <p>32 = Applies only to ETSI E1 lines (30B+D). Enables handling the differences between the newer QSIG standard (ETS 300-172) and other ETSI-based standards (ETS 300-102 and ETS 300-403) in the conversion of B-channel ID values into timeslot values:</p> <ul style="list-style-type: none"> <li>In 'regular ETSI' standards, the timeslot is identical to the B-channel ID value, and the range for both is 1 to 15 and 17 to 31. The D-channel is identified as channel-id #16 and carried into the timeslot #16.</li> <li>In newer QSIG standards, the channel-id range is 1 to 30, but the timeslot range is still 1 to 15 and 17 to 31. The D-channel is not identified as channel-id #16, but is still carried into the timeslot #16.</li> </ul> <p>When this bit is set, the channel ID #16 is considered as a valid B-channel ID, but timeslot values are converted to reflect the range 1 to 15 and 17 to 31. This is the new QSIG mode of operation. When this bit is not set (default), the channel_id #16 is not allowed, as for all ETSI-like standards.</p>
<b>ISDNOutCallsBehavior</b> [Outgoing Calls Behavior]	1024 = Numbering plan / type for T1 IP→Tel calling number are defined according to the manipulation tables or according to RPID header (default). Otherwise, the Plan / type for T1 calls are set according to the length of the calling number
<b>ISDNIBehavior_x</b> [Q.931 Layer Response Behavior]	Same as the description for parameter 'ISDNBehavior' for a specific trunk ID (x = 0 to 7)
<b>ISDNInCallsBehavior_x</b> [Incoming Calls Behavior]	Same as the description for parameter 'ISDNInCallsBehavior' for a specific trunk ID (x = 0 to 7)
<b>ISDNOutCallsBehavior_x</b> [Outgoing Calls Behavior]	Same as the description for parameter 'ISDNOutCallsBehavior' for a specific trunk ID (x = 0 to 7)

## 6.12 Channel Parameters

The Channel Parameters define the DTMF, fax and modem transfer modes. Refer to [Appendix D](#) on page 203 for a detailed description of Fax and Modem transfer modes; refer to [Section 8.2](#) on page 139 for detailed description on DTMF transport modes.

Note that the Default Channel Parameters are applied to all Mediant 2000 channels.



**Note:** In [Table 6-7](#), parameters in brackets are the format in the Embedded Web Server .

**Table 6-7: Channel Parameters (continues on pages 128 to 131)**

<i>ini</i> File Field Name Web Parameter Name *	Valid Range and Description
<b>DJBufMinDelay</b> [Dynamic Jitter Buffer Minimum Delay]	0 to 150 msec (default = 70) Dynamic Jitter Buffer Minimum Delay. <b>Note:</b> For more information on the Jitter Buffer, refer to <a href="#">Section 6.12.1</a> on page 132.
<b>DJBufOptFactor</b> [Dynamic Jitter Buffer Optimization Factor]	Dynamic Jitter Buffer frame error / delay optimization factor. You can enter a value from 0 to 13. The default factor is 7. <b>Note 1:</b> Set to 13 for data (fax & modem) calls. <b>Note 2:</b> For more information on the Jitter Buffer, refer to <a href="#">Section 6.12.1</a> on page 132.
<b>FaxTransportMode</b> [Fax Transport Mode]	Fax transport mode that the gateway uses. You can select: 0 = Disable. 1 = T.38 Relay (default). 2 = Bypass. <b>Note:</b> If parameter IsFaxUsed = 1, then FaxTransportMode is always set to 1 (T.38 relay).
<b>FaxRelayEnhancedRedundancyDepth</b> [Fax Relay Enhanced Redundancy Depth]	0 to 4 (default = 0) Number of repetitions applied to control packets when using T.38 standard.
<b>FaxRelayRedundancyDepth</b> [Fax Relay Redundancy Depth]	Number of times that each fax relay payload is retransmitted to the network. You can enter a value from 0 to 2. The default value is 0.
<b>FaxRelayMaxRate</b> [Fax Relay Max Rate (bps)]	Limits the maximum rate at which fax messages are transmitted. 0 = 2.4 kbps 1 = 4.8 kbps 2 = 7.2 kbps 3 = 9.6 kbps 4 = 12.0 kbps 5 = 14.4 kbps (default).
<b>FaxRelayECMEnable</b> [Fax Relay ECM Enable]	0 = Disable using ECM (Error Correction Mode) mode during fax relay. 1 = Enable using ECM mode during fax relay (default).
<b>FaxModemBypassCoderType</b> [Fax/Modem Bypass Coder Type]	Coder the gateway uses when performing fax/modem bypass. Usually, high-bit-rate coders such as G.711 should be used. You can select: 0 = G711 A-law 64 (default). 1 = G711 $\mu$ -law. 4 = G726 32. 11 = G726_40.

Table 6-7: Channel Parameters (continues on pages 128 to 131)

<i>ini</i> File Field Name Web Parameter Name *	Valid Range and Description
<b>CNGDetectorMode</b> [CNG Detector Mode]	0 = Disable (default). 1 = Event Only (N/A). 2 = Relay. T.38 fax relay session is initiated by the originating fax if 'IsFaxUsed = 1'. Note that using this mode isn't recommended.
<b>FaxModemBypassM</b> [Fax/Modem Bypass Packing Factor]	Number of (20 msec) coder payloads that are used to generate a fax/modem bypass packet. You can enter a value of 1, 2 or 3 coder payloads. The default value is 1 coder payload.
<b>FaxBypassPayloadType</b> [Fax Bypass Payload Type]	Determines the fax bypass RTP dynamic payload type. The valid range is 96 to 120. The default value is 102.
<b>ModemBypassPayloadType</b>	Modem Bypass dynamic payload type (range 0-127). The default value is 103.
<b>DetFaxOnAnswerTone</b> [Detect Fax on Answer Tone]	0 = Starts T.38 procedure on detection of V.21 preamble (default). 1 = Starts T.38 Procedure on detection of CED fax answering tone.
<b>FaxModemBypassBasicRTPPacketInterval</b>	0 = set internally (default) 1 = 5 msec (not recommended) 2 = 10 msec 3 = 20 msec
<b>FaxModemBypassDJBufMinDelay</b>	0 to 150 msec (default=40) Determines the Jitter Buffer delay during fax and modem bypass session
<b>NSEMode</b>	Cisco compatible fax and modem bypass mode 0 = NSE disabled (default) 1 = NSE enabled <b>Note 1:</b> This feature can be used only if VxxModemTransportType=2 (Bypass) <b>Note 2:</b> If NSE mode is enabled the SDP contains the following line: "a=rtmpmap:100 X-NSE/8000" <b>Note 3:</b> To use this feature: <ul style="list-style-type: none"> <li>The Cisco gateway must include the following definition: "modem passthrough nse payload-type 100 codec g711alaw".</li> <li>Set the Modem transport type to Bypass mode ('VxxModemTransportType = 2') for all modems.</li> <li>Configure the gateway parameter NSEPayloadType= 100</li> </ul> In NSE bypass mode the gateway starts using G.711 A-Law (default) or G.711 $\mu$ -Law, according to the parameter 'FaxModemBypassCoderType'. The payload type used with these G.711 coders is a standard one (8 for G.711 A-Law and 0 for G.711 $\mu$ -Law). The parameters defining payload type for the "old" AudioCodes' Bypass mode. 'FaxBypassPayloadType' and 'ModemBypassPayloadType' are not used with NSE Bypass. The bypass packet interval is selected according to the parameter 'FaxModemBypassBasicRtpPacketInterval'.
<b>NSEPayloadType</b>	NSE payload type for Cisco Bypass compatible mode. The valid range is 96-127. The default value is 105. <b>Note:</b> Cisco gateways usually use NSE payload type of 100.
<b>V22ModemTransportType</b> [V.22 Modem Transport Type]	V.22 Modem Transport Type that the gateway uses. You can select: 0 = Transparent 2 = Modem Bypass (default).
<b>V23ModemTransportType</b> [V.23 Modem Transport Type]	V.23 Modem Transport Type that the gateway uses. You can select: 0 = Transparent 2 = Modem Bypass (default).
<b>V32ModemTransportType</b> [V.32 Modem Transport Type]	V.32 Modem Transport Type that the gateway uses. You can select: 0 = Transparent 2 = Modem Bypass (default). <b>Note:</b> This option applies to V.32 and V.32bis modems.

**Table 6-7: Channel Parameters (continues on pages 128 to 131)**

<i>ini</i> File Field Name Web Parameter Name *	Valid Range and Description
<b>V34ModemTransportType</b> [V.34 Modem Transport Type]	V.34 Modem Transport Type that the gateway uses. You can select: 0 = Transparent 2 = Modem Bypass (default). <b>Note:</b> This option applies to V.34 and V.90 modems.
<b>InputGain</b> [Input Gain]	PCM input gain control in dB. This parameter sets the level for the received (PSTN→IP) signal. You can enter a value from -32 to 31 dB. The default value is 0 dB. <b>Note:</b> This parameter is intended for advanced users. Changing it affects other gateway functionalities.
<b>VoiceVolume</b> [Voice Volume]	Voice gain control in dB. This parameter sets the level for the transmitted (IP→PSTN) signal. You can enter a value from -32 to 31 dB. The default value is 0 dB.
<b>RTPRedundancyDepth</b> [RTP Redundancy Depth]	0 = Disable redundancy packets generation (default) 1 = Enable generation of RFC 2198 redundancy packets.
<b>RFC2198PayloadType</b>	RTP redundancy packet payload type, according to RFC 2198. The range is 96-127. The default is 104. Applicable if "RTPRedundancyDepth=1"
<b>EnableSilenceCompression</b> [Silence Suppression]	0 = Silence Suppression disabled (default). 1 = Silence Suppression enabled. 2 [Enable without adaptation] = A single silence packet is sent during silence period (applicable only to G.729). Silence Suppression is a method conserving bandwidth on VoIP calls by not sending packets when silence is detected. <b>Note:</b> If the selected coder is G.729, the following rules determine the value of the "annexb" parameter of the fmtp attribute in the SDP. EnableSilenceCompression = 0 → "annexb=no". EnableSilenceCompression = 1 → "annexb=yes". EnableSilenceCompression = 2 and IsCiscoSCEMode = 0 → "annexb=yes". EnableSilenceCompression = 2 and IsCiscoSCEMode = 1 → "annexb=no".
<b>IsCiscoSCEMode</b>	0 = There isn't a Cisco gateway at the remote side (default). 1 = There is a Cisco gateway at the remote side. When there is a Cisco gateway at the remote side, the local gateway must set the value of the "annexb" parameter of the fmtp attribute in the SDP to "no". This logic should be used if 'EnableSilenceCompression = 2' (enable without adaptation). In this case, Silence Suppression should be used on the channel but not declared in the SDP.
<b>EnableEchoCanceller</b> [Echo Canceler]	0 = Echo Canceler disabled. 1 = Echo Canceler Enabled (default).  <b>Note:</b> Refer also to the parameters 'MaxEchoCancellerLength' and 'EchoCancellerLength' (described in <a href="#">Table 6-1</a> on page 90).
<b>EnableStandardSIDPayloadType</b> [Enable RFC 3389 CN Payload Type]	0 = Disable (default). 1 = Enable. If enabled, the SID (comfort noise) packets are sent with the RTP SID payload type according to RFC 3389. Applicable to G.711 and G.726 coders. If disabled, the G.711 SID packets are sent in a proprietary method.
<b>DTMFVolume</b> [DTMF Volume]	-31 to 0 corresponding to -31 dBm to 0 dBm in 1 dB steps (default = -11 dBm) DTMF gain control.
<b>DTMFTransportType</b> [DTMF Transport Type]	0 = Erase digits from voice stream, do not relay to remote. 2 = Digits remain in voice stream. 3 = Erase digits from voice stream, relay to remote according to RFC 2833. <b>Note:</b> This parameter is automatically updated if one of the following parameters is configured: IsDTMFUsed, TxDTMFOption or RxDTMFOption.

Table 6-7: Channel Parameters (continues on pages 128 to 131)

<b>ini File Field Name Web Parameter Name *</b>	<b>Valid Range and Description</b>
<b>RFC2833PayloadType</b> [RFC 2833 Payload Type]	The RFC 2833 DTMF relay dynamic payload type. Range: 96 to 99, 106 to 127; Default = 96 The 100, 102 to 105 range is allocated for proprietary usage. Cisco is using payload type 101 for RFC 2833. <b>Note:</b> When RFC 2833 payload type (PT) negotiation is used (TxDTMFOption=4), this payload type is used for the received DTMF packets. If negotiation isn't used, this payload type is used for receive and for transmit.
<b>MGCPDTMFDetectionPoint</b>	0 = DTMF event is reported on the start of a detected DTMF digit. 1 = DTMF event is reported on the end of a detected DTMF digit (default). <b>Note:</b> The parameter is used for out-of-band dialing.
<b>DTMFInterDigitInterval</b>	Time in msec between generated DTMFs to PSTN side Default = 100 (msec)
<b>DTMFDigitLength</b>	Time in msec for generating of DTMF tone to PSTN side Default = 100 (msec)

## 6.12.1 Dynamic Jitter Buffer Operation

Voice frames are transmitted at a fixed rate. If the frames arrive at the other end at the same rate, voice quality is perceived as good. In many cases, however, some frames can arrive slightly faster or slower than the other frames. This is called jitter (delay variation), and degrades the perceived voice quality. To minimize this problem, the gateway uses a jitter buffer. The jitter buffer collects voice packets, stores them and sends them to the voice processor in evenly spaced intervals.

The Mediant 2000 uses a dynamic jitter buffer that can be configured using two parameters:

- Minimum delay, 'DJBufMinDelay' (0 msec to 150 msec). Defines the starting jitter capacity of the buffer. For example, at 0 msec, there is no buffering at the start. At the default level of 70 msec, the gateway always buffers incoming packets by at least 70 msec worth of voice frames.
- Optimization Factor, 'DJBufOptFactor' (0 to 12, 13). Defines how the jitter buffer tracks to changing network conditions. When set at its maximum value of 12, the dynamic buffer aggressively tracks changes in delay (based on packet loss statistics) to increase the size of the buffer and doesn't decay back down. This results in the best packet error performance, but at the cost of extra delay. At the minimum value of 0, the buffer tracks delays only to compensate for clock drift and quickly decays back to the minimum level. This optimizes the delay performance but at the expense of a higher error rate.

The default settings of 70 msec Minimum delay and 7 Optimization Factor should provide a good compromise between delay and error rate. The jitter buffer "holds" incoming packets for 70 msec before making them available for decoding into voice. The coder polls frames from the buffer at regular intervals to produce continuous speech. As long as delays in the network do not change (jitter) by more than 70 msec from one packet to the next, there is always a sample in the buffer for the coder to use. If there is more than 70 msec of delay at any time during the call, the packet arrives too late. The coder tries to access a frame and is not able to find one. The coder must produce a voice sample even if a frame is not available. It therefore compensates for the missing packet by adding a Bad-Frame-Interpolation (BFI) packet. This loss is then flagged as the buffer being too small. The dynamic algorithm then causes the size of the buffer to increase for the next voice session. The size of the buffer may decrease again if the gateway notices that the buffer is not filling up as much as expected. At no time does the buffer decrease to less than the minimum size configured by the Minimum delay parameter.

### Special Optimization Factor Value: 13

One of the purposes of the Jitter Buffer mechanism is to compensate for clock drift. If the two sides of the VoIP call are not synchronized to the same clock source, one RTP source generates packets at a lower rate, causing under-runs at the remote Jitter Buffer. In normal operation (optimization factor 0 to 12), the Jitter Buffer mechanism detects and compensates for the clock drift by occasionally dropping a voice packet or by adding a BFI packet.

Fax and modem devices are sensitive to small packet losses or to added BFI packets. Therefore to achieve better performance during modem and fax calls, the Optimization Factor should be set to 13. In this special mode the clock drift correction is performed less frequently - only when the Jitter Buffer is completely empty or completely full. When such condition occurs, the correction is performed by dropping several voice packets simultaneously or by adding several BFI packets simultaneously, so that the Jitter Buffer returns to its normal condition.

## 6.13 Configuration Files Parameters

The configuration files (Call Progress Tones, PRT, Voice Prompts and CAS) can be loaded to the Mediant 2000 via the Embedded Web Server (refer to Section 5.11.2 on page 82), or via TFTP session.

➤ **To load the configuration files via TFTP, take these 3 steps:**

1. In the *ini* file, define the files to be loaded to the device. You can also define in the *ini* file whether the loaded files should be stored in the non-volatile memory so that the TFTP process is not required every time the device boots up.
2. Locate the configuration files you want to load and the *ini* file in the same directory.
3. Invoke a BootP/TFTP session; the *ini* and configuration files are loaded onto the device.

Table 6-8 below describes the *ini* file parameters that are associated with the configuration files.

**Table 6-8: Configuration File Parameters**

<i>ini</i> File Field Name	Valid Range and Description
<b>CallProgressTonesFilename</b>	The name of the file containing the Call Progress Tones definitions. Refer to Section 7 for additional information on how to create and load this file.
<b>VoicePromptsFileName</b>	The name (and path) of the file containing the Voice Prompts definitions. Refer to Section 7.2 on page 137 for additional information on how to create and load this file.
<b>CASfilename</b>	This is the name of the file containing specific CAS protocol definition (such as 'E_M_WinkTable.dat'). These files are provided to support various types of CAS signaling.
<b>CASfilename_x</b>	It is possible to load up to 8 different CAS files (x=0 to 7), by repeating the CASFileName parameter. Each CAS file can be associated with one or more of the gateway trunks, using "CASTableIndex_x" parameter.
<b>CASTablesNum</b>	Number, 1 to 8. Specifies how many CAS configuration files are loaded.
<b>PrerecordedTonesFileName</b>	The name (and path) of the file containing the Prerecorded Tones.
<b>SaveConfiguration</b>	Set to 1 to store the CPT, PRT, CAS and Voice Prompts files in the non-volatile memory.

## Reader's Notes

# 7 Configuration Files

This section describes the configuration (*dat*) files that are load (in addition to the *ini* file) to the gateway. The configuration files are:

- Call Progress Tones file (refer to Section 7.1 below).
- Prerecorded Tones file (refer to Section 7.2 on page 137).
- Voice Prompts file (refer to Section 7.3 on page 137).
- CAS protocol configuration files (refer to Section 7.4 on page 138).

To load any of the configuration files to the Mediant 2000 use the Embedded Web Server (refer to Section 5.11.2 on page 82) or alternatively specify the name of the relevant configuration file in the gateway's *ini* file and load it (the *ini* file) to the gateway (refer to Section B.6 on page 190).

## 7.1 Configuring the Call Progress Tones

The Call Progress Tones, configuration file used by the Mediant 2000 is a binary file (with the extension *dat*) which contains the definitions of the Call Progress Tones (levels and frequencies) that are detected / generated by the Mediant 2000.

Users can either use, one of the supplied Mediant 2000 configuration (*dat*) files, or construct their own file. To construct their own configuration file, users are recommended, to modify the supplied *usa\_tone.ini* file (in any standard text editor) to suit their specific requirements, and to convert it (the modified *ini* file) into binary format using the "TrunkPack Downloadable Conversion Utility" supplied with the software package. For the description of the procedure on how to convert CPT *ini* file to a binary *dat* file, refer to Section G.1.1 on page 214.

Note that only the *dat* file can be loaded to the Mediant 2000 gateway.

To load the Call Progress Tones (*dat*) file to the Mediant 2000, use the Embedded Web Server (refer to Section 5.11.2 on page 82) or the *ini* file (refer to Section 6.12.1 on page 132).

### 7.1.1 Format of the Call Progress Tones Section in the *ini* File

Using the CPT section of this configuration file, the User can create up to 16 different Call Progress Tones using up to 15 different frequencies (in the range of 300 Hz to 1980 Hz). Each of these Call Progress Tones is specified by its tone frequency (either single or dual frequencies are supported) and its tone cadence. The tone cadence is specified by 2 sets of on/off periods (you can discard the use of the first on/off cycle by setting the relevant parameters to zero). When a tone is composed of a single frequency, the second frequency field must be set to zero.

For a continuous tone (such as dial tone), only the "First Signal On time" should be specified. In this case, the parameter specifies the detection period. For example, if it equals 300, the tone is detected after 3 seconds (300 x 10 msec). The minimum detection time is 100 msec.

Users can specify several tones of the same type. These additional tones are used only for tone detection. Generation of a specific tone conforms to the first definition of the specific tone. For example, Users can define an additional dial tone by appending the second dial tone's definition lines to the first tone definition in the *ini* file. The Mediant 2000 reports dial tone detection if either of the two tones is detected.

The Call Progress Tones section of the *ini* file format starts from the following string:

- **[NUMBER OF CALL PROGRESS TONES]** – Contains the following key:
  - "Number of Call Progress Tones" defining the number of Call Progress Tones that are defined in the file.

- **[CALL PROGRESS TONE #X]** – containing the Xth tone definition (starting from 1 and not exceeding the number of Call Progress Tones defined in the first section) using the following keys:
  - **Tone Type** – Call Progress Tone type

**Figure 7-1: Call Progress Tone Types**

- |                                |
|--------------------------------|
| 1. Dial Tone                   |
| 2. Ringback Tone               |
| 3. Busy Tone                   |
| 7. Reorder Tone                |
| 17. Call Waiting Ringback Tone |
| 23. Hold Tone                  |

- **Low Freq [Hz]** – Frequency in hertz of the lower tone component in case of dual frequency tone, or the frequency of the tone in case of single tone.
- **High Freq [Hz]** – Frequency in hertz of the higher tone component in case of dual frequency tone, or zero (0) in case of single tone.
- **Low Freq Level [-dBm]** – Generation level 0 dBm to –31 dBm in [dBm].
- **High Freq Level** – Generation level. 0 to –31 dBm. The value should be set to ‘32’ in the case of a single tone.
- **First Signal On Time [10 msec]** – “Signal On” period (in 10 msec units) for the first cadence on-off cycle.
- **First Signal Off Time [10 msec]** – “Signal Off” period (in 10 msec units) for the first cadence on-off cycle.
- **Second Signal On Time [10 msec]** – “Signal On” period (in 10 msec units) for the second cadence on-off cycle.
- **Second Signal Off Time [10 msec]** – “Signal Off” period (in 10 msec units) for the second cadence on-off cycle.
- **Default Duration [msec]** - The default duration (in 1 msec units) of the generated tone.



- Note 1:** When the same frequency is used for a continuous tone and a cadence tone, the ‘Signal On Time’ parameter of the continuous tone must have a value that is greater than the ‘Signal On Time’ parameter of the cadence tone. Otherwise the continuous tone is detected instead of the cadence tone.
- Note 2:** The tones frequency should differ by at least 40 Hz from one tone to other defined tones.

For example: to configure the dial tone to 440 Hz only, define the following text:

**Figure 7-2: Defining a Dial Tone Example**

```
#Dial tone
[CALL PROGRESS TONE #1]
Tone Type=1
Low Freq [Hz]=440
High Freq [Hz]=0
Low Freq Level [-dBm]=10 (-10 dBm)
High Freq Level [-dBm]=32 (use 32 only if a single tone is required)
First Signal On Time [10msec]=300; the dial tone is detected after 3 sec
First Signal Off Time [10msec]=0
Second Signal On Time [10msec]=0
Second Signal Off Time [10msec]=0
```

## 7.2 Prerecorded Tones (PRT) File

The Call Progress Tones mechanism has several limitations, such as a limited number of predefined tones and a limited number of frequency integrations in one tone. To work around these limitations and provide tone generation capability that is more flexible, the PRT file can be used. If a specific prerecorded tone exists in the PRT file, it takes precedence over the same tone that exists in the CPT file and is played instead of it.

Note that the prerecorded tones are used only for generation of tones. Detection of tones is performed according to the CPT file.

### 7.2.1 PRT File Format

The PRT *dat* file contains a set of prerecorded tones to be played by the Mediant 2000 during operation. Up to 40 tones (totaling approximately 10 minutes) can be stored in a single file in flash memory. The prerecorded tones (raw data PCM or L8 files) are prepared offline using standard recording utilities (such as CoolEdit™) and combined into a single file using the TrunkPack Downloadable Conversion utility (refer to Section G.1.4 on page 218).

The raw data files must be recorded with the following characteristics:

- Coders: G.711 A-law, G.711  $\mu$ -law or Linear PCM
- Rate: 8 kHz
- Resolution: 8-bit
- Channels: mono

The generated PRT file can then be loaded to the Mediant 2000 using the BootP/TFTP utility (refer to Section 6.13 on page 133) or via the Embedded Web Server (Section 5.11.2 on page 82).

The prerecorded tones are played repeatedly. This enables you to record only part of the tone and play it for the full duration. For example, if a tone has a cadence of 2 seconds on and 4 seconds off, the recorded file should contain only these 6 seconds. The PRT module repeatedly plays this cadence for the configured duration. Similarly, a continuous tone can be played by repeating only part of it.

## 7.3 Voice Prompts File



**Note:** The Voice Prompts file is applicable only to the VXML application.

The voice announcement file contains a set of Voice Prompts to be played by the Mediant 2000 during operation. The voice announcements are prepared offline using standard recording utilities and combined into a single file using the TrunkPack Downloadable Conversion Utility.

The generated announcement file can then be loaded to the Mediant 2000 using the BootP/TFTP utility (refer to Section 6.12.1 on page 132) or via the Embedded Web Server (Section 5.11.2 on page 82).

If the size of the combined Voice Prompts file is less than 1 MB, it can permanently be stored in flash memory. Larger files, up to 10 MB, are stored in RAM, and should be loaded again (using BootP/TFTP utility) after the Mediant 2000 is reset.

The Voice Prompts integrated file is a collection of raw voice recordings and / or *wav* files. These recordings can be prepared using standard utilities, such as CoolEdit, Goldwave™ and others. The raw voice recordings must be sampled at 8000 kHz / mono / 8 bit. The *wav* files must be recorded with G.711 $\mu$ -Law/A-Law/Linear.

When the list of recorded files is converted to a single *voiceprompts.dat* file, every Voice Prompt is tagged with an ID number, starting with "1". This ID is used later by the Mediant 2000 to start playing the correct announcement. Up to 1000 Voice Prompts can be used.



**Note:** The Voice Prompt ID is used in the VXML file to specify the message that is to be played.

AudioCodes provides a professionally recorded English (U.S.) Voice Prompts file.

➤ **To generate and load the Voice Prompts file, take these 3 steps:**

1. Prepare one or more voice files using standard utilities.
2. Use the TrunkPack Downloadable Conversion Utility to generate the *voiceprompts.dat* file from the pre-recorded voice messages (refer to Section [G.1.2](#) on page [215](#)).
3. Load the *voiceprompts.dat* file to the Mediant 2000 either by using a TFTP procedure (refer to Section [6.12.1](#) on page [132](#)), or via the Embedded Web Server (Section [5.11.2](#) on page [82](#)).

## 7.4 CAS Protocol Configuration Files

The CAS Protocol Configuration Files contain the CAS Protocol definitions to be used for CAS-terminated trunks. Users can either use the files supplied or construct their own files.

It is possible to load up to eight files and to use different files for different trunks.

Note that all CAS files loaded together must belong to the same Trunk Type (either E1 or T1).

## 8 Gateway Capabilities Description

### 8.1 Proxy or Registrar Registration Example

```
REGISTER sip:servername SIP/2.0
VIA: SIP/2.0/UDP 212.179.22.229;branch=z9hG4bRaC7AU234
From: <sip:GWRegistrationName@sipgatewayname>;tag=1c29347
To: <sip:GWRegistrationName@sipgatewayname>
Call-ID: 10453@212.179.22.229
Seq: 1 REGISTER
Expires: 3600
Contact: sip:GWRegistrationName@212.179.22.229
Content-Length: 0
```

The "**servername**" string is defined according to the following rules:

- The "**servername**" is equal to "RegistrarName" if configured. The "RegistrarName" can be any string.
- Otherwise, the "**servername**" is equal to "RegistrarIP" (either FQDN or numerical IP address), if configured.
- Otherwise the "**servername**" is equal to "ProxyName" if configured. The "ProxyName" can be any string.
- Otherwise the "**servername**" is equal to "ProxyIP" (either FQDN or numerical IP address).

The parameter 'GWRegistrationName' can be any string. If the parameter is not defined, the parameter 'UserName' is used instead.

The "**sipgatewayname**" parameter (defined in the *ini* file or set from the Web browser), can be any string. Some Proxy servers require that the "**sipgatewayname**" (in Register messages) is set equal to the Registrar / Proxy IP address or to the Registrar / Proxy domain name.

The Register message is sent to the Registrar's IP address (if configured) or to the Proxy's IP address. The message is sent once per gateway. The registration request is resent according to the parameter 'RegistrartionTimeDivider'. For example, if 'RegistrationTimeDivider = 70' (%) and Registration Expires time = 3600, the gateway resends its registration request after  $3600 \times 70\% = 2520$  sec. The default value of 'RegistrartionTimeDivider' is 50%.

### 8.2 Redirect Number and Calling Name (Display)

The following tables define the Mediant 2000 redirect number and calling name (Display) support for various PRI variants:

**Table 8-1: Calling Name (Display)**

	DMS-100	NI-2	4/5ESS	Euro ISDN
NT→TE	Yes	Yes	No	Yes
TE→NT	Yes	Yes	No	No

**Table 8-2: Redirect Number**

	DMS-100	NI-2	4/5ESS	Euro ISDN
NT→TE	Yes	Yes	Yes	Yes
TE→NT	Yes	Yes	Yes	No

## 8.3 ISDN Overlap Dialing

Overlap dialing is a dialing scheme used by several ISDN variants to send and / or receive called number digits one right after the other (or several at a time). As opposed to the enbloc dialing scheme in which a complete number is sent.

The Mediant 2000 can optionally support ISDN overlap dialing for incoming ISDN calls for the entire gateway by setting 'ISDNRxOverlap' to 1, or per E1/T1 span by setting 'ISDNRxOverlap\_x' to 1 ('x' represents the number of the trunk, 0 to 7).

To play a Dial tone to the ISDN user side when an empty called number is received, set 'ISDNINCallsBehavior = 65536' (bit #16) causing the Progress Indicator to be included in the SetupAck ISDN message.

The Mediant 2000 stops collecting digits (for ISDN→IP calls) when:

- The sending device transmits a "sending complete" IE in the ISDN Setup or the following Info messages to signal that no more digits are going to be sent.
- The inter-digit timeout (configured by the parameter 'TimeBetweenDigits') expires. The default for this timeout is 4 seconds.
- The maximum allowed number of digits (configured by the parameter 'MaxDigits') is reached. The default is 30 digits.

Relevant parameters (described in [Table 6-6](#) on page 122):

- ISDNRxOverlap
- ISDNRxOverlap\_x
- TimeBetweenDigits
- MaxDigits
- ISDNINCallsBehavior

## 8.4 Using ISDN NFAS

In regular (non-NFAS) T1 ISDN trunks, a single 64 kbps channel carries signaling for the other 23 B-channels of that particular T1 trunk. This channel is called the D-channel and usually resides on timeslot # 24.

The ISDN Non-Facility Associated Signaling (NFAS) feature enables use of a single D-channel to control multiple PRI interfaces.

With NFAS it is possible to define a group of T1 trunks, called an NFAS group, in which a single D-channel carries ISDN signaling messages for the entire group. The NFAS group's B-channels are used to carry traffic, such as voice or data. The NFAS mechanism also enables definition of a backup D-channel on a different T1 trunk, to be used if the primary D-channel fails.

The NFAS group comprises several T1 trunks. Each T1 trunk is called an 'NFAS member'. The T1 trunk whose D-channel is used for signaling is called the 'Primary NFAS Trunk'. The T1 trunk whose D-channel is used for backup signaling is called the 'Backup NFAS Trunk'. The primary and backup trunks each carry 23 B-channels while all other NFAS trunks each carry 24 B-channels.

The Mediant 2000 supports multiple NFAS groups. Each group should contain different T1 trunks.

The NFAS group is identified by an NFAS GroupID number (possible values are 1, 2, 3 and 4). To assign a number of T1 trunks to the same NFAS group, use the parameter 'NFASGroupNumber\_x = groupID'. 'x' stands for the physical trunkID (0 to 7).

The parameter 'DchConfig\_x = Trunk\_type' is used to define the type of NFAS trunk. Trunk\_type is set to 0 for the primary trunk, to 1 for the backup trunk and to 2 for an ordinary NFAS trunk. 'x' stands for the physical trunkID (0 to 7).

For example, to assign the first four Mediant 2000 T1 trunks to NFAS group #1, in which trunk #0 is the primary trunk and trunk #1 is the backup trunk, use the following configuration:

```
NFASGroupNumber_0 = 1
NFASGroupNumber_1 = 1
NFASGroupNumber_2 = 1
NFASGroupNumber_3 = 1
DchConfig_0 = 0           ;Primary T1 trunk
DchConfig_1 = 1           ;Backup T1 trunk
DchConfig_2 = 2           ;24 B-channel NFAS trunk
DchConfig_3 = 2           ;24 B-channel NFAS trunk
```

The NFAS parameters are described in [Table 6-6](#) on page 122.



**Note:** In the current version the NFAS parameters cannot be configured via the 'Trunk Settings' screen in the Embedded Web Server. Use *ini* file configuration instead.

### 8.4.1 NFAS Interface ID

Several ISDN switches require an additional configuration parameter per T1 trunk, that is called 'Interface Identifier'. In NFAS T1 trunks the Interface Identifier is sent explicitly in Q.931 Setup / Channel Identification IE for all NFAS trunks, except for the B-channels of the Primary trunk (refer to note 1 below).

The Interface ID can be defined per each member (T1 trunk) of the NFAS group, and must be coordinated with the configuration of the Switch.

The default value of the Interface ID is identical to the number of the physical T1 trunk (0 for the first Mediant 2000 trunk, 1 for the second Mediant 2000 T1 trunk etc. up to 7).

To define an explicit Interface ID for a T1 trunk (that is different from the default), use the following parameters:

- ISDNBehavior\_x = 512 (x = 0 to 7 identifying the Mediant 2000 physical trunk)
- ISDNNFASInterfaceID\_x = ID (x = 0 to 255)



**Note 1:** Usually the Interface Identifier is included in the Q.931 Setup/Channel Identification IE only on T1 trunks that doesn't contain the D-channel. Calls initiated on B-channels of the Primary T1 trunk, by default, don't contain the Interface Identifier. Setting the parameter 'ISDNBehavior\_x' to 2048' forces the inclusion of the Channel Identifier parameter also for the Primary trunk.

**Note 2:** The parameter 'ISDNNFASInterfaceID\_x = ID' can define the 'Interface ID' for any Primary T1 trunk, even if the T1 trunk is not a part of an NFAS group. However, to include the Interface Identifier in Q.931 Setup/Channel Identification IE configure 'ISDNBehavior\_x = 2048' in the *ini* file.

## 8.4.2 Working with DMS-100 Switches

The DMS-100 switch requires the following NFAS Interface ID definitions:

- InterfaceID #0 for the Primary trunk
- InterfaceID #1 for the Backup trunk (refer to the note below)
- InterfaceID #2 for a 24 B-channel T1 trunk
- InterfaceID #3 for a 24 B-channel T1 trunk
- Etc.

**Note:** In the current version, the Mediant 2000 doesn't support the DMS-100 Backup trunk. Therefore, InterfaceID #1, should not be used.

For example, if four T1 trunks on a Mediant 2000 are configured as a single NFAS group that is used with a DMS-100 switch, the following parameters should be used:

```
ISDNNFASInterfaceID_0 = 0
ISDNNFASInterfaceID_1 = 2
ISDNNFASInterfaceID_2 = 3
ISDNNFASInterfaceID_3 = 4
NFASGroupNumber_0 = 1
NFASGroupNumber_1 = 1
NFASGroupNumber_2 = 1
NFASGroupNumber_3 = 1
DchConfig_0 = 0 ;Primary T1 trunk
DchConfig_2 = 2 ;24 B-channel NFAS trunk
DchConfig_3 = 2 ;24 B-channel NFAS trunk
DchConfig_4 = 2 ;24 B-channel NFAS trunk
```

## 8.5 Configuring the DTMF Transport Types

You can control the way DTMF digits are transported over the IP network to the remote endpoint. The following five modes are supported:

- Using INFO message according to the Nortel IETF draft:  
In this mode DTMF digits are carried to the remote side within INFO messages.  
To enable this mode set:
  - 'IsDTMFUsed = 1' (Protocol Management>Protocol Defenition>DTMF & Dialing>Use Out-of-Band DTMF = Yes)
  - 'OutOfBandDTMFFormat = 1' (Protocol Management>Protocol Defenition>DTMF & Dialing>Out-of-Band DTMF Format = INFO (Nortel))
  - 'RxDTMFOption = 0' (Protocol Management>Protocol Defenition>DTMF & Dialing>Declare RFC 2833 in SDP = No)

Note that in this mode DTMF digits are erased from the audio stream (DTMFTransportType is automatically set to 0).

- Using INFO message according to Cisco's style:  
In this mode DTMF digits are carried to the remote side within INFO messages.  
To enable this mode set:
  - 'IsDTMFUsed = 1' (Use Out-of-Band DTMF = Yes)
  - 'OutOfBandDTMFFormat = 2' (Out-of-Band DTMF Format = INFO (Cisco))
  - 'RxDTMFOption = 0' (Declare RFC 2833 in SDP = No)

Note that in this mode DTMF digits are erased from the audio stream (DTMFTransportType is automatically set to 0).

- Using NOTIFY messages according to <draft-mahy-sipping-signaled-digits-01.txt>:  
In this mode DTMF digits are carried to the remote side using NOTIFY messages.  
To enable this mode set:
  - 'IsDTMFUsed = 1' (Use Out-of-Band DTMF = Yes)
  - 'OutOfBandDTMFFormat = 3' (Out-of-Band DTMF Format = NOTIFY)
  - 'RxDTMFOption = 0' (Declare RFC 2833 in SDP = No)

Note that in this mode DTMF digits are erased from the audio stream (DTMFTransportType is automatically set to 0).

- Using RFC 2833 relay with Payload type negotiation:  
In this mode, DTMF digits are carried to the remote side as part of the RTP stream in accordance with RFC 2833 standard.  
To enable this mode set:
  - 'IsDTMFUsed = 0' (Use Out-of-Band DTMF = No)
  - 'TxDTMFOption = 4' (Protocol Management>Protocol Defenition>DTMF & Dialing> DTMF RFC 2833 Negotiation = Enable)
  - 'RxDTMFOption = 3' (Declare RFC 2833 in SDP = Yes)
  - 'DTMFTransportType = 3' (Advanced Configuration>Channel Settings>Voice Settings>DTMF Transport Type = RFC 2833 Relay DTMF)

Note that to set the RFC 2833 payload type with a different value (other than its default, 96) configure the 'RFC2833PayloadType' parameter. The gateway negotiates the RFC 2833 payload type using local and remote SDP and sends packets using the PT from the received SDP. The gateway expects to receive RFC 2833 packets with the same PT as configured by the 'RFC2833PayloadType' parameter. The RFC 2833 packets are sent even if the remote side didn't include the send "telephone-event" parameter in its SDP, in which case the gateway uses the same PT for send and for receive.

5. Sending DTMF digits (in RTP packets) as part of the audio stream (DTMF Relay is disabled): Note that this method is normally used with G.711 coders; with other Low Bit Rate (LBR) coders the quality of the DTMF digits is reduced.

To set this mode:

- 'IsDTMFUsed = 0' (Use Out-of-Band DTMF = No)
- 'TxDTMFOption = 0' (DTMF RFC 2833 Negotiation = Disable)
- 'RxDTMFOption = 0' (Declare RFC 2833 in SDP = No)
- 'DTMFTransportType = 2' (DTMF Transport Type = Transparent DTMF)



**Note 1:** The gateway is always ready to receive DTMF packets over IP, in all possible transport modes: INFO messages, Notify and RFC 2833 (in proper payload type) or as part of the audio stream.

**Note 2:** To exclude RFC 2833 Telephony event parameter from the gateway's SDP, set 'RxDTMFOption = 0' in the *ini* file.

The following parameters affect the way the Mediant 2000 SIP handles the DTMF digits:

**Table 8-3: Summary of DTMF Configuration Parameters (continues on pages 144 to 145)**

<i>ini</i> File Field Name [Web Name]	Valid Range and Description
<b>IsDTMFUsed</b> [Use Out-of-Band DTMF]	Use out-of-band signaling to relay DTMF digits. 0 = Disable, DTMF digits are sent according to DTMFTransportType parameter. (default) 1 = Enable sending DTMF digits within INFO or NOTIFY messages.  <b>Note:</b> When out-of-band DTMF transfer is used, DTMFTransportType is automatically set to 0 (erase the DTMF digits from the RTP stream).
<b>OutOfBandDTMFFormat</b> [Out-of-Band DTMF Format]	The exact method to send out-of-band DTMF digits 1 = INFO format (Nortel) 2 = INFO format (Cisco) - (default) 3 = NOTIFY format <draft-mahy-sipping-signaled-digits-01.txt>  <b>Note 1:</b> To use out-of-band DTMF, set "IsDTMFUsed=1". <b>Note 2:</b> When using out-of-band DTMF, the "DTMFTransportType" parameter is automatically set to 0, to erase the DTMF digits from RTP path.
<b>TxDTMFOption</b> [DTMF RFC 2833 Negotiation]	0 = No negotiation, DTMF digit is sent according to the parameters 'DTMFTransportType' and 'RFC2833PayloadType'. 4 = Enable RFC 2833 payload type (PT) negotiation  <b>Note 1:</b> This parameter is applicable only if "IsDTMFUsed=0" (out of-band DTMF is not used) <b>Note 2:</b> If enabled, the gateway: <ul style="list-style-type: none"> <li>• Negotiates RFC 2833 payload type using local and remote SDPs.</li> <li>• Sends DTMF packets using RFC 2833 PT according to the PT in the received SDP.</li> <li>• Expects to receive RFC 2833 packets with the same PT as configured by the "RFC2833PayloadType" parameter.</li> </ul> <b>Note 3:</b> If the remote party doesn't include the RFC 2833 DTMF relay payload type in the SDP, the gateway uses the same PT for send and for receive. <b>Note 4:</b> If TxDTMFOption is set to 0, the RFC 2833 payload type is set according to the parameter 'RFC2833PayloadType' for both transmit and receive.

Table 8-3: Summary of DTMF Configuration Parameters (continues on pages 144 to 145)

<i>ini</i> File Field Name [Web Name]	Valid Range and Description
<b>RxDTMFOption</b>	<p>Defines the supported Receive DTMF negotiation method.</p> <p>0 = Don't declare RFC 2833 Telephony-event parameter in SDP            1 = n/a            2 = n/a            3 = Declare RFC 2833 "Telephony-event" parameter in SDP (default)</p> <p>The gateway is designed to always be receptive to RFC 2833 DTMF relay packets. Therefore, it is always correct to include the "Telephony-event" parameter as a default in the SDP. However some gateways use the absence of the "telephony-event" from the SDP to decide to send DTMF digits inband using G.711 coder, if this is the case you can set "RxDTMFOption=0".</p>
<b>RFC2833PayloadType</b>	<p>The RFC 2833 DTMF relay dynamic payload type.</p> <p>Range: 96 to 99, 106 to 127; Default = 96            The 100, 102 to 105 range is allocated for proprietary usage.            Cisco is using payload type 101 for RFC 2833.</p> <p><b>Note:</b> When RFC 2833 payload type (PT) negotiation is used (TxDTMFOption=4), this payload type is used for the received DTMF packets. If negotiation isn't used, this payload type is used for receive and for transmit.</p>
<b>MGCPDTMFDetectionPon</b>	<p>0 = Send out of-band DTMF message on starting point of DTMF digit            1 = Send DTMF message on ending point of DTMF digit (default)</p>
<b>DTMFDigitLength</b>	<p>Time in msec for generating DTMF tones to the PSTN side (if received in INFO).            The default value is 100 msec.</p>
<b>DTMFInterDigitInterval</b>	<p>Time in msec between generated DTMFs to PSTN side (if received in INFO).            The default value is = 100 msec.</p>
<b>DTMFVolume</b> [DTMF Volume]	<p>DTMF level for regenerated digits to PSTN side (-31 to 0, corresponding to -31 dBm to 0 dBm in 1 dB steps, default = -11 dBm)</p>
<b>DTMFTransportType</b> [DTMF Transport Type]	<p>0 = Erase digits from voice stream, do not relay to remote.            2 = Digits remain in voice stream.            3 = Erase digits from voice stream, relay to remote according to RFC 2833.</p> <p><b>Note:</b> This parameter is automatically updated if one of the following parameters is configured: IsDTMFUsed, TxDTMFOption or RxDTMFOption.</p>

## 8.6 Configuring the Gateway's Alternative Routing (based on Connectivity and QoS)

The Alternative Routing feature enables reliable routing of Tel to IP calls when Proxy isn't used. The Mediant 2000 gateway periodically checks the availability of connectivity and suitable Quality of Service (QoS) before routing. If the expected quality cannot be achieved, an alternative IP route for the prefix (phone number) is selected.

Note that if the alternative routing destination is the gateway itself, the call can be configured to be routed back to one of the gateway's trunk groups and thus back into the PSTN (PSTN Fallback).

### 8.6.1 Alternative Routing Mechanism

When a Tel→IP call is routed through the Mediant 2000 gateway, the call's destination number is compared to the list of prefixes defined in the Tel to IP Routing table (described in Section 5.8.4.1 on page 49). The Tel to IP Routing table is scanned for the destination number's prefix starting at the top of the table. When an appropriate entry (destination number matches one of the prefixes) is found; the prefix's corresponding destination IP address is checked. If the destination IP address is disallowed, an alternative route is searched for in the following table entries.

Destination IP address is disallowed if no ping to the destination is available (ping is continuously initiated every 7 seconds), when an inappropriate level of QoS was detected, or when DNS host name is not resolved. The QoS level is calculated according to delay or packet loss of previously ended calls. If no call statistics are received for two minutes, the QoS information is reset.

The Mediant 2000 gateway matches the rules starting at the top of the table. For this reason, enter the main IP route above any alternative route.

### 8.6.2 Determining the Availability of Destination IP Addresses

To determine the availability of each destination IP address (or host name) in the routing table, one (or all) of the following (configurable) methods are applied:

- Connectivity – The destination IP address is queried periodically (currently only by ping).
- QoS – The QoS of an IP connection is determined according to RTCP (Real-Time Control Protocol) statistics of previous calls. Network delay (in msec) and network packet loss (in percentage) are separately quantified and compared to a certain (configurable) threshold. If the calculated amounts (of delay or packet loss) exceed these thresholds the IP connection is disallowed.
- DNS resolution – When host name is used (instead of IP address) for the destination route, it is resolved to an IP address by a DNS server. Connectivity and QoS are then applied to the resolved IP address.

### 8.6.3 PSTN Fallback as a Special Case of Alternative Routing

The purpose of the PSTN Fallback feature is to enable the Mediant 2000 gateway to redirect PSTN originated calls back to the legacy PSTN network if a destination IP route is found unsuitable (disallowed) for voice traffic at a specific time.

To enable PSTN fallback, assign the IP address of the gateway itself as an alternative route to the desired prefixes. Note that calls (now referred to as IP to Tel calls) can be re-routed to a specific trunk group using the Routing parameters.

## 8.6.4 Relevant Parameters

The following parameters (described in [Table 6-5](#)) are used to configure the Alternative Routing mechanism:

- AltRoutingTel2IPEnable
- AltRoutingTel2IPMode
- IPConnQoSMaxAllowedPL
- IPConnQoSMaxAllowedDelay

## 8.7 Working with Supplementary Services

The Mediant 2000 SIP gateway supports the following supplementary services:

- Hold / Retrieve
- Transfer
- Call Forward (doesn't initiate call forward, only responds to call forward request)
- Call Waiting

Mediant 2000 SIP users are only required to enable the Hold and Transfer features. The call forward (supporting 30x redirecting responses) and call waiting (receive of 182 response) features are enabled by default. Note that all call participants must support the specific used method.



**Note:** When working with application servers (such as BroadSoft's BroadWorks) in client server mode (the application server controls all supplementary services and keypad features by itself), the gateway's supplementary services must be disabled.

### 8.7.1 Call Hold and Retrieve Features

- The party that initiates the hold is called the holding party, the other party is called the held party. The Mediant 2000 can't initiate the hold, but it can respond to hold request, and as such it is a held party.
- After a successful hold, the held party should hear HELD\_TONE, defined in gateway's Call Progress Tones file.
- Retrieve can be performed only by the holding party while the call is held and active.
- After a successful retrieve the voice should be connected again.
- The hold and retrieve functionalities are implemented by Reinvite messages. The IP address 0.0.0.0 as the connection IP address or the string "a=inactive" in the received Reinvite SDP cause the gateway to enter Hold state and to play held tone (configured in the gateway) to the PBX/PSTN. If the string "a=recvonly" is received in the SDP message, the gateway stops sending RTP packets, but continues to listen to the incoming RTP packets. Usually, the remote party plays, in this scenario, Music on-hold (MOH) and the Mediant 2000 forwards the MOH to the held party.

### 8.7.2 Call Transfer

There are two types of call transfers:

- Consultation Transfer
- Blind Transfer

The common way to perform a consultation transfer is as follows:

In the transfer scenario there are three parties:

Party A - transferring, Party B – transferred, Party C – transferred to.

- A Calls B.
- B answers.
- A presses the hookflash and puts B on-hold (party B hears a hold tone)
- A dials C.
- After A completed dialing C, A can perform the transfer by on-hooking the A phone.
- After the transfer is completed, B and C parties are engaged in a call.

The transfer can be initiated at any of the following stages of the call between A to C:

- Just after completing dialing C phone number - Transfer from setup.
- While hearing ring back – Transfer from alert.
- While speaking to C – Transfer from active.

Blind transfer is performed after we have a call between A and B, and A party decides to transfer the call to C immediately without speaking with C.

The result of the transfer is a call between B and C (just like consultation transfer only skipping the consultation stage).

The Mediant 2000 doesn't initiate call transfer, it only can respond to call transfer request.

## 8.8 TDM Tunneling

The Mediant 2000 TDM Tunneling feature allows you to tunnel groups of digital trunk spans or timeslots (B-channels) over the IP network. TDM Tunneling utilizes the internal routing capabilities of the Mediant 2000 (working without Proxy control) to receive voice and data streams from TDM (1 to 16 E1/T1/J1) spans or individual timeslots, convert them into packets and transmit them automatically over the IP network (using point-to-point or point-to-multipoint gateway distributions). A Mediant 2000 opposite it (or several Mediant 2000 gateways, when point-to-multipoint distributions is used) converts the IP packets back into TDM traffic. Each timeslot can be targeted to any other timeslot within a trunk in the opposite Mediant 2000.

### 8.8.1 Implementation

When TDM Tunneling is enabled ('EnableTDMOverIP' is set to 1 on the originating Mediant 2000), the originating Mediant 2000 automatically initiates SIP calls from all enabled B-channels belonging to the E1/T1/J1 spans that are configured with the 'Transparent' protocol. The called number of each call is the internal phone number of the B-channel that the call originates from. The IP to Trunk Group routing table is used to define the destination IP address of the terminating Mediant 2000. The terminating Mediant 2000 gateway automatically answers these calls if its E1/T1 protocol is set to 'Transparent' (ProtocolType = 5) and parameter 'ChannelSelectMode = 0' (By Phone Number).

**Note:** It is possible to configure both gateways to also operate in symmetric mode. To do so, set 'EnableTDMOverIP' to 1 and configure the Tel to IP Routing tables in both Mediant 2000 gateways. In this mode, each gateway (after it is reset) initiates calls to the second gateway. The first call for each B-channel is answered by the second gateway.

The Mediant 2000 monitors the established connections continuously, if for some reason one or more calls are released, the gateway automatically reestablishes these "broken" connections. In addition, when a failure in a physical trunk or in the IP network occurs, the Mediant 2000 gateways reestablish the tunneling connections as soon as the network restores.

**Note:** It is recommended to use the keep-alive mechanism for each connection by activating "session expires" timeout, and using Reinvite messages.

By utilizing the 'Profiles' mechanism (refer to Section 5.8.5 on page 55) you can configure the TDM Tunneling feature to choose different settings, based on a timeslot or groups of timeslots. For example, you can use low-bit-rate vocoders to transport voice, and 'Transparent' coder to transport data (e.g., for D-channel). You can also use Profiles to assign ToS (for DiffServ) per source, a time-slot carrying data or signaling gets a higher priority value than a time-slot carrying voice.

For tunneling of E1/T1 CAS trunks enable RFC 2833 CAS relay mode (CASTransportType = 1).

Figure 8-1 and Figure 8-2 show an example of *ini* files for two Mediant 2000 gateways implementing TDM Tunneling for four E1 spans. Note that in this example both gateways are dedicated to TDM tunneling.

**Figure 8-1: ini File Example for TDM Tunneling (Originating Side)**

```

EnableTDMOverIP = 1

;E1_TRANSPARENT_31
ProtocolType_0 = 5
ProtocolType_1 = 5
ProtocolType_2 = 5
ProtocolType_3 = 5

prefix = '*',10.8.24.12' ;(IP address of the Mediant 2000 in the opposite location)

; Channel selection by Phone number
ChannelSelectMode = 0

;Profiles can be used do define different coders per B-channels, such as Transparent
; coder for B-channels (time slot 16) that carries PRI signaling.
TrunkGroup = 0/1-31,1000,1
TrunkGroup = 1/1-31,2000,1
TrunkGroup = 2/1-31,3000,1
TrunkGroup = 3/1-31,4000,1
TrunkGroup = 0/16-16,7000,2
TrunkGroup = 1/16-16,7001,2
TrunkGroup = 2/16-16,7002,2
TrunkGroup = 3/16-16,7003,2

CoderName = 'g7231'
CoderName = 'Transparent'

CoderName_1 = 'g7231'
CoderName_2 = 'Transparent'

TelProfile_1 = voice,$$,1,$$, $$,$$, $$,$$, $$,$$, $$,$$
TelProfile_2 = data,$$,2,$$, $$,$$, $$,$$, $$,$$, $$,$$
    
```

**Figure 8-2: ini File Example for TDM Tunneling (Terminating Side)**

```

;E1_TRANSPARENT_31
ProtocolType_0 = 5
ProtocolType_1 = 5
ProtocolType_2 = 5
ProtocolType_3 = 5

; Channel selection by Phone number
ChannelSelectMode = 0

TrunkGroup = 0/1-31,1000,1
TrunkGroup = 1/1-31,2000,1
TrunkGroup = 2/1-31,3000,1
TrunkGroup = 3/1-31,4000,1
TrunkGroup = 0/16-16,7000,2
TrunkGroup = 1/16-16,7001,2
TrunkGroup = 2/16-16,7002,2
TrunkGroup = 3/16-16,7003,2

CoderName = 'g7231'
CoderName = 'Transparent'

CoderName_1 = 'g7231'
CoderName_2 = 'Transparent'

TelProfile_1 = voice,$$,1,$$, $$,$$, $$,$$, $$,$$, $$,$$
TelProfile_2 = data,$$,2,$$, $$,$$, $$,$$, $$,$$, $$,$$
    
```

## 8.9 Call Detail Report

The Call Detail Report (CDR) contains vital statistic information on calls made by the gateway. CDRs are generated at the end and (optionally) at the beginning of each call (determined by the parameter 'CDRReportLevel'). The destination IP address for CDR logs is determined by the parameter 'CDRSyslogServerIP'.

The following CDR fields are supported:

**Table 8-4: Supported CDR Fields**

Field Name	Description
Cid	Board's Logic Channel Number
CallId	H.323/SIP Call Identifier
Trunk	Physical Trunk Number
BChan	Selected B-Channel
ConId	H.323/SIP Conference ID
TG	Trunk Group Number
EPTyp	Endpoint Type
Orig	Call Originator (IP, Tel)
Sourcelp	Source IP Address
DestIp	Destination IP Address
TON	Source Phone Number Type
NPI	Source Phone Number Plan
SrcPhoneNum	Source Phone Number
TON	Destination Phone Number Type
NPI	Destination Phone Number Plan
DstPhoneNum	Destination Phone Number
DstNumBeforeMap	Destination Number Before Manipulation
Durat	Call Duration
Coder	Selected Coder
Intrv	Packet Interval
Rtplp	RTP IP Address
Port	Remote RTP Port
TrmSd	Initiator of Call Release (IP, Tel, Unknown)
TrmReason	Termination Reason
Fax	Fax Transaction during the Call
InPackets	Number of Incoming Packets
OutPackets	Number of Outgoing Packets
PackLoss	Number of Outgoing Lost Packets
Uniqueld	unique RTP ID
SetupTime	Call Setup Time
ConnectTime	Call Connect Time
ReleaseTime	Call Release Time
RTPdelay	RTP Delay
RTPjitter	RTP Jitter
RTPssrc	Local RTP SSRC
RemoteRTPssrc	Remote RTP SSRC
RedirectReason	Redirect Reason
TON	Redirection Phone Number Type
NPI	Redirection Phone Number Plan
RedirectPhonNum	Redirection Phone Number

## 8.10 Trunk to Trunk Routing Example

This example describes two Mediant 2000 gateways, each interface with the PSTN through four E1 spans. Gateway "A" is configured to route all incoming Tel→IP calls to gateway "B". Gateway "B" generates calls to PSTN on the same E1 Trunk as the call was originally received (in gateway "A").

Gateway "A" IP address is 192.168.3.50

Gateway "B" IP address is 192.168.3.51

### **Ini File Parameters of Gateways "A" and "B":**

1. Define, for both gateways, four trunk groups; each with 30 B-channels:
  - TrunkGroup\_1 = 0/1-31,1000
  - TrunkGroup\_2 = 1/1-31,2000
  - TrunkGroup\_3 = 2/1-31,3000
  - TrunkGroup\_4 = 3/1-31,4000
2. In gateway "A", add the originating Trunk Group ID, as a prefix, to the destination number, for Tel→IP calls:
  - AddTrunkGroupAsPrefix=1
3. In gateway "A", route all incoming PSTN calls, starting with the prefixes 1, 2, 3 and 4, to gateway's "B" IP address:
  - Prefix = 1, 192.168.3.51
  - Prefix = 2, 192.168.3.51
  - Prefix = 3, 192.168.3.51
  - Prefix = 4, 192.168.3.51

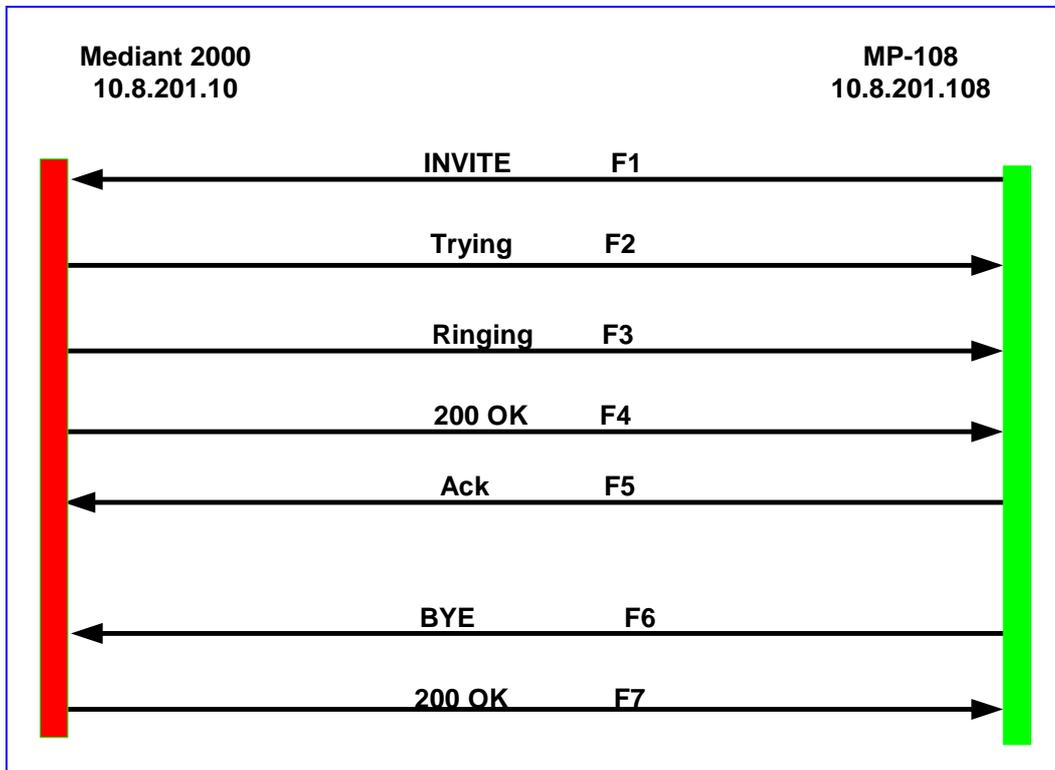
Note: It is also possible to define "Prefix = \*,192.168.3.51" instead of the four lines above.
4. In gateway "B", route IP→PSTN calls to Trunk Group ID according to the first digit of the called number:
  - PSTNPrefix = 1,1
  - PSTNPrefix = 2,2
  - PSTNPrefix = 3,4
  - PSTNPrefix = 4,4
5. In gateway "B", remove the first digit from each IP→PSTN number, before it is used in an outgoing call:
  - NumberMapIP2Tel = \*,1

## 8.11 SIP Call Flow Example

The Call Flow, shown in [Figure 8-3](#), describes SIP messages exchanged between Mediant 2000 gateway and an MP-108 gateway during a simple call.

**MP-108** with phone number "8000", calls Mediant 2000 with phone number "1000":

**Figure 8-3: SIP Call Flow Example**



### F1 10.8.201.108 ==> 10.8.201.10 INVITE

```

INVITE sip:1000@10.8.201.10;user=phone SIP/2.0
Via: SIP/2.0/UDP 10.8.201.108;branch=z9hG4bKacsiJkDGd
From: <sip:8000@10.8.201.108>;tag=1c5354
To: <sip:1000@10.8.201.10>
Call-ID: 534366556655skKw-8000--1000@10.8.201.108
CSeq: 18153 INVITE
Contact: <sip:8000@10.8.201.108;user=phone>
User-Agent: Audiocodes-Sip-Gateway/MP-108 FXS/v.4.20.299.410
Supported: 100rel,em
Accept-Language: en
Allow: REGISTER,OPTIONS,INVITE,ACK,CANCEL,BYE,NOTIFY,PRACK,REFER,INFO
Content-Type: application/sdp
Content-Length: 208

v=0
o=AudiocodesGW 18132 74003 IN IP4 10.8.201.108
s=Phone-Call
c=IN IP4 10.8.201.108
t=0 0
m=audio 4000 RTP/AVP 8 96
a=rtpmap:8 pcma/8000
a=rtpmap:96 telephone-event/8000
a=fmtp:96 0-15
a=ptime:20
  
```

### F2 10.8.201.10 ==> 10.8.201.108 Trying

```
SIP/2.0 100 Trying
Via: SIP/2.0/UDP 10.8.201.108;branch=z9hG4bKacsiJkDGd
From: <sip:8000@10.8.201.108>;tag=1c5354
To: <sip:1000@10.8.201.10>
Call-ID: 534366556655skKw-8000--1000@10.8.201.108
Server: Audiocodes-Sip-Gateway/TrunkPack 1610/v.4.20.299.412
CSeq: 18153 INVITE
Content-Length: 0
```

### F3 10.8.201.10 ==> 10.8.201.108 180 Ringing

```
SIP/2.0 180 Ringing
Via: SIP/2.0/UDP 10.8.201.108;branch=z9hG4bKacsiJkDGd
From: <sip:8000@10.8.201.108>;tag=1c5354
To: <sip:1000@10.8.201.10>;tag=1c7345
Call-ID: 534366556655skKw-8000--1000@10.8.201.108
Server: Audiocodes-Sip-Gateway/TrunkPack 1610/v.4.20.299.412
CSeq: 18153 INVITE
Supported: 100rel,em
Content-Length: 0
```



**Note:** Phone "1000" answers the call, and sends "200 OK" message to MP gateway 10.8.201.108.

### F4 10.8.201.10 ==> 10.8.201.108 200 OK

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP 10.8.201.108;branch=z9hG4bKacsiJkDGd
From: <sip:8000@10.8.201.108>;tag=1c5354
To: <sip:1000@10.8.201.10>;tag=1c7345
Call-ID: 534366556655skKw-8000--1000@10.8.201.108
CSeq: 18153 INVITE
Contact: <sip:1000@10.8.201.10;user=phone>
Server: Audiocodes-Sip-Gateway/TrunkPack 1610/v.4.20.299.412
Supported: 100rel,em
Allow: REGISTER,OPTIONS,INVITE,ACK,CANCEL,BYE,NOTIFY,PRACK,REFER,INFO
Content-Type: application/sdp
Content-Length: 206

v=0
o=AudiocodesGW 30221 87035 IN IP4 10.8.201.10
s=Phone-Call
c=IN IP4 10.8.201.10
t=0 0
m=audio 7210 RTP/AVP 8 96
a=rtpmap:8 pcma/8000
a=ptime:20
a=rtpmap:96 telephone-event/8000
a=fmtp:96 0-15
```

### F5 10.8.201.108 ==> 10.8.201.10 ACK

```
ACK sip:1000@10.8.201.10;user=phone SIP/2.0
Via: SIP/2.0/UDP 10.8.201.108;branch=z9hG4bKacZYpJWxZ
From: <sip:8000@10.8.201.108>;tag=1c5354
To: <sip:1000@10.8.201.10>;tag=1c7345
Call-ID: 534366556655skKw-8000--1000@10.8.201.108
User-Agent: Audiocodes-Sip-Gateway/MP-108 FXS/v.4.20.299.410
```

```
CSeq: 18153 ACK
Supported: 100rel,em
Content-Length: 0
```



**Note:** Phone "8000" goes on-hook; gateway 10.8.201.108 sends "BYE" to gateway 10.8.201.10. Voice path is established.

#### F6 10.8.201.108 ==> 10.8.201.10 BYE

```
BYE sip:1000@10.8.201.10;user=phone SIP/2.0
Via: SIP/2.0/UDP 10.8.201.108;branch=z9hG4bKacRKCVBud
From: <sip:8000@10.8.201.108>;tag=1c5354
To: <sip:1000@10.8.201.10>;tag=1c7345
Call-ID: 534366556655skKw-8000--1000@10.8.201.108
User-Agent: Audiocodes-Sip-Gateway/MP-108 FXS/v.4.20.299.410
CSeq: 18154 BYE
Supported: 100rel,em
Content-Length: 0F7 10.2.37.10 ==> 10.2.37.20      200 OK
```

#### F7 10.8.201.10 ==> 10.8.201.108 200 OK

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP 10.8.201.108;branch=z9hG4bKacRKCVBud
From: <sip:8000@10.8.201.108>;tag=1c5354
To: <sip:1000@10.8.201.10>;tag=1c7345
Call-ID: 534366556655skKw-8000--1000@10.8.201.108
Server: Audiocodes-Sip-Gateway/TrunkPack 1610/v.4.20.299.412
CSeq: 18154 BYE
Supported: 100rel,em
Content-Length: 0
```

## 8.12 SIP Authentication Example

Mediant 2000 gateway supports basic and digest authentication types, according to SIP RFC 3261 standard. A proxy server might require authentication before forwarding an INVITE message. A Registrar/Proxy server may also require authentication for client registration. A proxy replies to an unauthenticated INVITE with a 407 Proxy Authorization Required response, containing a Proxy-Authenticate header with the form of the challenge. After sending an ACK for the 407, the User Agent can then resend the INVITE with a Proxy-Authorization header containing the credentials.

User Agent, Redirect or Registrar servers typically use 401 Unauthorized responses to challenge authentication containing a WWW-Authenticate header, and expect the re-INVITE to contain an Authorization header.

The following example describes the Digest Authentication procedure including computation of User Agent credentials.

The REGISTER request is sent to Registrar/Proxy server for registration, as follows:

```
REGISTER sip:10.2.2.222 SIP/2.0
Via: SIP/2.0/UDP 10.1.1.200
From: <sip:122@10.1.1.200>;tag=1c17940
To: <sip:122@10.1.1.200>
Call-ID: 634293194@10.1.1.200
User-Agent: Audiocodes-Sip-Gateway/TrunkPack 1610/v.4.20.299.412
CSeq: 1 REGISTER
Contact: sip:122@10.1.1.200:
Expires:3600
```

On receiving this request the Registrar/Proxy returns 401 Unauthorized response.

```
SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP 10.2.1.200
From: <sip:122@10.2.2.222 >;tag=1c17940
To: <sip:122@10.2.2.222 >
Call-ID: 634293194@10.1.1.200
Cseq: 1 REGISTER
Date: Mon, 30 Jul 2001 15:33:54 GMT
Server: Columbia-SIP-Server/1.17
Content-Length: 0
WWW-Authenticate: Digest realm="audiocodes.com",
nonce="11432d6bce58ddf02e3b5e1c77c010d2",
stale=FALSE,
algorithm=MD5
```

According to the sub-header present in the WWW-Authenticate header the correct REGISTER request is formed.

Since the algorithm used is MD5, take:

The username from the ini file: M2K-AudioCodes

The realm return by the proxy: audiocodes.com

The password from the ini file: AudioCodes.

The equation to be evaluated: (according to RFC this part is called A1).

**“M2K-AudioCodes:audiocodes.com:AudioCodes”.**

The MD5 algorithm is run on this equation and stored for future usage.

The result is: “a8f17d4b41ab8dab6c95d3c14e34a9e1”

Next we need to evaluate the par called A2. We take:

The method type “REGISTER”

Using SIP protocol "sip"

Proxy IP from ini file "10.2.2.222"

The equation to be evaluated:

**"REGISTER:sip:10.2.2.222".**

The MD5 algorithm is run on this equation and stored for future usage.

The result is:"a9a031cfddcb10d91c8e7b4926086f7e"

The final stage:

The A1 result

The nonce from the proxy response: "11432d6bce58ddf02e3b5e1c77c010d2"

The A2 result

The equation to be evaluated:

**"A1:11432d6bce58ddf02e3b5e1c77c010d2:A2".**

The MD5 algorithm is run on this equation. The outcome of the calculation is the response needed by the gateway to be able to register with the Proxy.

The response is: "b9c45d0234a5abf5ddf5c704029b38cf"

At this time a new REGISTER request is issued with the response:

```
REGISTER sip:10.2.2.222 SIP/2.0
Via: SIP/2.0/UDP 10.1.1.200
From: <sip: 122@10.1.1.200>;tag=1c23940
To: <sip: 122@10.1.1.200>
Call-ID: 654982194@10.1.1.200
Server: Audiocodes-Sip-Gateway/TrunkPack 1610/v.4.20.299.412
CSeq: 1 REGISTER
Contact: sip:122@10.1.1.200:
Expires:3600
Authorization: Digest, Username: MP108-AudioCodes,
realm="audiocodes.com",
nonce="11432d6bce58ddf02e3b5e1c77c010d2",
uri="10.2.2.222",
response=" b9c45d0234a5abf5ddf5c704029b38cf"
```

On receiving this request, if accepted by the Proxy, the proxy returns a 200 OK response closing the REGISTER transaction.

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP 10.1.1.200
From: <sip: 122@10.1.1.200>;tag=1c23940
To: <sip: 122@10.1.1.200>
Call-ID: 654982194@10.1.1.200
Cseq: 1 REGISTER
Date: Thu, 26 Jul 2001 09:34:42 GMT
Server: Columbia-SIP-Server/1.17
Content-Length: 0
Contact: <sip:122@10.1.1.200>; expires="Thu, 26 Jul 2001 10:34:42 GMT"; action=proxy;
q=1.00
Contact: <122@10.1.1.200:>; expires="Tue, 19 Jan 2038 03:14:07 GMT"; action=proxy;
q=0.00
Expires: Thu, 26 Jul 2001 10:34:42 GMT
```

## 8.13 Nortel IMS Specific Features and Configuration

### 8.13.1 SIP2PRI Gateway

- To enable Nortel's IMS SIP2PRI gateway specific features, add the following parameters to the *ini* file:

```

ApplicationProfile = 4444
IsProxyUsed = 1
SendInviteToProxy = 1
ProxyIP = <Proxy IP address>
EnableHold = 1
EnableTransfer = 1
EnableForward = 1
xferPrefix = $
RemovePrefix = 1
EnableRPIHeader = 1
IsDTMFUsed = 1
OutOfBandDTMFFormat = 1
DTMFTransportType = 0
AddTrunkGroupAsPrefix = 1
NumberMapTel2IP = *, 1
AlwaysUseRouteTable = 1
RouteModeIP2Tel = 0
DefaultNumber = 9999000
SourceNumberMapTel2IP = 9999,0,Anonymous,0
AddIEinSetup = <IE data in (hex format), used in ISDN SETUP>
SendIEonTG = <list of Trunk Group IDs, from where the IE is sent>
    
```



**Note:** The parameter 'ApplicationProfile = 4444' enables Nortel's ISDN PRI specific features.

#### 8.13.1.1 SIP to PRI Calls

Routing IP-originated calls to PRI is performed according to a concatenation of domain name and trunk group name. Before applying any of the routing or manipulation rules on an incoming IP call, the SIP2PRI gateway creates a new number from the SIP INVITE message; combining the domain name (marked in red) followed by a forward slash '/' and trunk group name (marked in blue), finally appended with the initial phone number (marked in pink). The routing rules are applied only after the new number is created.

For example:

```

INVITE sip:2145551234@nortelnetworks.com:5060;norteltrkgrp=TrkGrp3;user=phone
SIP/2.0"
    
```

From the above SIP INVITE message, the following called number is created:

```
nortelnetworks.com/TrkGrp32145551234
```



**Note 1:** 'norteltrkgrp' is a hard-coded string that always precedes the trunk group name.

**Note 1:** Different domain and trunk group names are supported.

For the SIP2PRI gateway to be able to apply the routing rules on the combined string, the prefix (domain name and trunk group) must previously be defined in the IP to Trunk Group Routing table either using the Embedded Web Server (refer to screen below) or via the *ini* file:

**Figure 8-4: IP to Trunk Group Routing Table**

	Dest. Phone Prefix	Source Phone Prefix	Source IP Address	Trunk Group ID
1	nortelnetworks.com/Trk	*	*	2

Via the *ini* file:

```
PSTNPrefix = nortelnetworks.com/TrkGrp3,2
```

Thus, in the above example, *any number* starting with the prefix 'nortelnetworks.com/TrkGrp3' is routed to trunk group number 2.

Using the routing table, any combination of domain name and trunk group name can be routed to any trunk group.



**Note:** To route calls to the trunk group before modifying the called number, add the parameter 'RouteModelIP2Tel = 0' to the *ini* file.

After applying the routing rule (the trunk to which the call is to be routed is already determined), it is possible to modify the combined called number (before it is sent to the PRI ISDN) to its initial state by removing the created prefix:

- Using the *ini* file parameter 'RemovePrefix = 1'; the called number is automatically modified (by removal of the prefix) to the original called number '2145551234'.
- Setting 'RemovePrefix = 0' and using the number manipulation rules (either via the Embedded Web Server or the *ini* file as described below):

```
NumberMapIP2Tel = nortelnetworks.com/TrkGrp3,26
```

### 8.13.1.2 PRI to SIP Calls

Trunk group ID and domain name are added to the generated INVITE messages for ISDN→IP calls.

For example:

```
INVITE sip:8795551212@nortelnetworks.com:5060;norteltrkgrp=5;user=phone
SIP/2.0
```

The PRI2SIP gateway sends SIP INVITE message with a trunk group number (marked in blue), preceded by 'norteltrkgrp' (hard-coded string) and domain name (marked in red).

The **trunk group** number is automatically determined as the trunk group from which the call was received (for detailed information on trunk groups, refer to Section 5.8.4.2 on page 51).



**Note:** Usually each E1/T1 span is configured as a separate trunk group.

Different Domain names, according to the originated trunk group (or E1/T1 spans), can be determined via the Tel to IP Routing table.

To enable different domain names add the following parameters to the *ini* file:

```
AlwaysUseRouteTable = 1
IsProxyUsed = 1
AddTrunkGroupAsPrefix = 1
NumberMapTel2IP = *, 1
```

The parameter 'AddTrunkGroupAsPrefix' is set to differentiate incoming calls, based on the trunk from where the call arrived (it is assumed that each trunk group is composed of a single trunk) and use the Tel to IP Routing table or the Prefix parameter in the *ini* file to link each trunk to a different domain name. Finally, 'NumberMapTel2IP' is set to remove the leftmost digit from any called number (the previously added trunk group ID).

For example:

```
TrunkGroup_1 = 0/1-24
TrunkGroup_2 = 1/1-24
TrunkGroup_3 = 2/1-24

Prefix = 1, example1.domain
Prefix = 2, example2.domain
Prefix = 3, example3.domain
```

In the example above, all calls that are originated in trunk 1 (associated with trunk group number 1) are sent to example1.domain. An incoming ISDN call, from trunk number 1, with called number 1234000, generates the following SIP message:

```
INVITE sip:1234000@example1.domain;norteltrkgrp=1;
```

### 8.13.1.3 Support for RPI Header

Support for Remote Party ID (RPI) header containing proprietary RPI-TON, RPI-NPI, privacy and screening parameters (EnableRPIHeader = 1).

#### Configuration of NPI/TON

Configuration of the calling number's NPI/TON values for outgoing IP→ISDN calls. If the calling number's NPI/TON values aren't provided in the Remote-Party-ID (RPID) header (received with the INVITE message) and are also not configured in the source number IP to Tel Manipulation table, then the calling number NPI/TON is set equal to the called number's NPI/TON.

NPI/TON values for the called number are defined in the RPID header or set by Number manipulation rules. If both are defined, the latter takes precedence over the former.

➤ **To configure a specific called number NPI/TON according to domain name and trunk group, take these 2 steps:**

1. Define RemovePrefix = 0, forcing the called number to include, as prefix, the Domain name and Trunk Group (this prefix can be used to define specific NPI/TON values), for example:

```
nortelnetworks.com/TrunkGroup32145551234
```

2. Use the number manipulation table (exemplified below) to assign NPI/TON values according to the number's prefix:

```
NumberMapIP2TEL= nortelnetworks.com/TrunkGroup3,30,$$, $$,1,2
```

In the above example, all numbers starting with the 'nortelnetworks.com/TrunkGroup3' string, are modified; the first 30 characters in this prefix string (domain and trunk group names) are erased, and the NPI/TON are set to 1/2 respectively.



**Note:** The \$\$ signs represent the empty parameters 'Prefix to add' and 'Number of digits to leave'.

#### 8.13.1.4 Transfer

To differentiate transferred calls (using the REFER method) from incoming ISDN calls, add the following parameter to the *ini* file: `xferPrefix = $`; the parameter adds the '\$' sign (or any other string) as a prefix to the transferred called number. The added string can be later removed using the number manipulation rule.



**Note:** When using the PRI2SIP gateway, the `NumberMapTel2IP = *, 1` manipulation rule is used. This manipulation rule also applies to transferred calls, corrupting the number by removing its first digit. Therefore, use the `xferPrefix = $` parameter to prepend a single character to the transferred number so that the manipulation table only removes the added character without altering the original number.

#### 8.13.1.5 Other Nortel Specific Parameters

If the parameter 'EarlyAnswerTimeout' > 0 and Q.931 CONNECT was not received from ISDN for 'EarlyAnswerTimeout', the Mediant 2000 responds with 200 OK.

Asserted-Identity header is ignored, if received by Mediant 2000 gateway in INVITE message and if 'ApplicationProfile = 4444'.

Mediant 2000 gateway doesn't include the Called Number Remoty-Party-ID header in INVITE (for ISDN→IP calls) if 'ApplicationProfile = 4444'.

## 8.13.2 SIP2CAS (Call Pilot) Gateway

➤ **To enable SIP2CAS gateway specific features, take these 2 steps:**

1. Add the following parameters to the *ini* file:

```

ApplicationProfile=1
IsProxyUsed = 1
SendInviteToProxy = 1
ProxyIP = <Proxy IP address>
EnableHold = 1
EnableTransfer = 1
EnableForward = 1
EnableNortelHeader = 1
ProtocolType = 2
IsDTMFUsed = 1
OutOfBandDTMFFormat = 1
DTMFTransportType = 0
TimeForReorderTone = 10
PlayRBToneonXfer = 1
    
```

2. Load the specific Ground Start CAS .dat file (supplied with the software package).

### 8.13.2.1 Supported Features

- Proprietary header in SIP 180 Ringing response, containing information about the trunk's physical port and timeslot which were seized to place the call. For example:

```

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP 10.8.1.15
From: sip:600@10.8.1.15;tag=1c14356
To: sip:1000@10.8.8.52;tag=1c29285
Call-ID: call-973660899-24@10.8.1.15
CSeq: 1 INVITE Supported: 100rel,em
Channel: interface=0;timeslot=1
Content-Length: 0
    
```

- When hook-flash is detected, the CAS gateway plays a dial tone and collects DTMF digits. The call is then transferred using a SIP REFER message:

```

REFER sip:600@10.8.1.15;user=phone SIP/2.0
Via: SIP/2.0/UDP 10.8.8.52;branch=z9hG4bKacdLIaZOe
From: sip:1000@10.8.8.52;tag=1c16026
To: <sip:600@10.8.1.15>;tag=1c10947
Call-ID: call-973573243-1@10.8.1.15
CSeq: 36026 REFER
Contact: 1001 <sip:1001@10.8.8.52>
Supported: 100rel,em
Max-Forwards: 70
Refer-To: sip:500@10.8.8.61
Referred-By: sip:1001@10.8.8.52
Content-Length: 0
    
```

- If the call transfer fails, the gateway either plays a busy tone (if the destination client to whom the call is transferred is busy) or it plays a reorder tone (for all other reasons). The tone is played for 10 seconds (`TimeForReorderTone=10`) towards the CallPilot. Usually, the CallPilot detects these Call Progress Tones and either disconnects the call or unholds it using double hook-flash signaling. If the CallPilot doesn't react to these tones, the gateway releases the call, and idle the CAS A/B bit state.
- If the call transfer fails (if user B is busy, for instance), detection of a double hook-flash from the CAS side unholds the call and retrieves the original call (using `reINVITE`).
- If the call transfer succeeds, the gateway plays a Ringback tone for 'TimeForReorder' (about 10 seconds) and tears down the call. To play the Ringback tone set `'PlayRBToneonXfer=1'`.
- Disabling the B-channel – If DTMF digits aren't received from the CallPilot after applying a dial tone to the CAS side (usually for 30 seconds, i.e., the duration of a dial tone), the gateway's B-channel converts to 'Block' state. During 'Block' state, the channel is put out of service and is not used for IP→Tel calls. To enable the B-channel, apply CAS A/B idle state (0,1) from the CallPilot side.

### 8.13.3 DTMF Configuration

To use out of band SIP INFO messages to relay DTMF digits, add the following parameters to the *ini* file.

```
IsDTMFUsed = 1
OutOfBandDTMFFormat = 1
RxDTMFOption = 0
```

To use RFC 2833 DTMF relay, add the following parameters to the *ini* file.

```
IsDTMFUsed = 0
TxDTMFOption = 4
RxDTMFOption = 3
DTMFTransportType = 3
```

For detailed information on DTMF transport modes, refer to Section 8.5 on page 143.

## Reader's Notes

## 9 Diagnostics

Several diagnostic tools are provided, enabling you to identify correct functioning of the Mediant 2000, or an error condition with a probable cause and a solution or workaround.

- Front panel indicator LEDs on the Mediant 2000. The location and functionality of the front panel LEDs is shown in Section 2.3.2 on page 23.
- Mediant 2000 Self-Testing on hardware initialization (refer to Section 9.1 below).
- Syslog Event Notification Messages (refer to Section 9.2 below).

### 9.1 Mediant 2000 Self-Testing

The Mediant 2000 features two self-testing modes: rapid and detailed.

**Rapid self-test mode** is invoked each time the media gateway completes the initialization process. This is a short test phase in which the only error detected and reported is failure in initializing hardware components. All Status and Error reports in this self-test phase are reported through Network Interface ports, as well as indicated by the LED Status Indicators.

**Detailed self-test mode** is invoked when initialization of the media gateway is completed and if the configuration parameter EnableDiagnostics is set to 1 (this parameter can be configured through the *ini* file mechanism). In this mode, the media gateway tests all the hardware components (memory, DSP, etc.), outputs the status of the test results, and ends the test. To continue operational running, reset the media gateway again but this time configure the EnableDiagnostics parameter to 0.

### 9.2 Syslog Support

Syslog protocol is an event notification protocol that enables a machine to send event notification messages across IP networks to event message collectors -also known as Syslog servers. Syslog protocol is defined in the IETF RFC 3164 standard.

Since each process, application and operating system was written independently, there is little uniformity to Syslog messages. For this reason, no assumption is made on the contents of the messages other than the minimum requirements of its priority.

Syslog uses UDP as its underlying transport layer mechanism. The UDP port that was assigned to Syslog is 514

The Syslog message is transmitted as an ASCII (American Standard Code for Information Interchange) message. The message starts with a leading "<" ('less-than' character), followed by a number, which is followed by a ">" ('greater-than' character). This is optionally followed by a single ASCII space.

The number described above is known as the Priority and represents both the Facility and Severity as described below. The Priority number consists of one, two, or three decimal integers.

For example:

```
<37> Oct 11 16:00:15 mymachine su: 'su root' failed for lonvick on /dev/pts/8
```

Note that when NTP is enabled, a timestamp string [hour:minutes:seconds] is added to all Syslog messages (for information on NTP, refer to Section 5.9.1.3 on page 63).

## 9.2.1 Syslog Servers

Users can use the provided Syslog server (ACSyslog08.exe) or other third-party Syslog servers.

Examples of Syslog servers available as shareware on the Internet:

- Kiwi Enterprises: <http://www.kiwisyslog.com/>
- The US CMS Server: [http://uscms.fnal.gov/hanlon/uscms\\_server/](http://uscms.fnal.gov/hanlon/uscms_server/)
- TriAction Software: <http://www.triaction.nl/Products/SyslogDaemon.asp>
- Netal SL4NT 2.1 Syslog Daemon: <http://www.netal.com>

A typical Syslog server application enables filtering of the messages according to priority, IP sender address, time, date, etc.

## 9.2.2 Operation

### 9.2.2.1 Sending the Syslog Messages

The Syslog client, embedded in the firmware of the Mediant 2000, sends error reports/events generated by the Mediant 2000 unit application to a Syslog server, using IP/UDP protocol.

### 9.2.2.2 Setting the Syslog Server

➤ **To set the Syslog server:**

- Use the Mediant 2000 Embedded Web Server (Advanced Configuration>Network Settings>screen section Syslog Settings) to enable the Syslog Server (Enable Syslog) and to enter its IP address (Syslog Server IP address); refer to [Figure 9-1](#) below.

**Figure 9-1: Setting the Syslog Server IP Address**

Syslog Settings	
Syslog Server IP Address	10.13.2.95
Enable Syslog	Enable

- Alternately, use the Embedded Web Server or the BootP/TFTP utility to load the *ini* configuration file containing both the IP address and the enabling parameters: SyslogServerIP and EnableSyslog respectively. For detailed information on the BootP/TFTP utility, refer to [Appendix B](#) on page 189. For an *ini* file example showing these parameters, refer to [Section 9.2.2.3](#) and to [Figure 9-2](#) under it.

### 9.2.2.3 The *ini* File Example for Syslog

[Figure 9-2](#) shows an *ini* file section with an example configuration for the address parameter SyslogServerIP and an example configuration for the client activation parameter EnableSyslog.

**Figure 9-2: The *ini* File Example for Syslog**

```
[Syslog]
SyslogServerIP = 10.2.0.136
EnableSyslog = 1
GWDebugLevel = 5
```

# 10 BootP/DHCP Support

## 10.1 Startup Process

The startup process (illustrated in [Figure 10-1](#) on page 168) begins when the gateway is reset (physically or from the Web / SNMP) and ends when the operational software is running. In the startup process, the network parameters, software and configuration files are obtained.

After the gateway powers up or after it is physically reset, it broadcasts a BootRequest message to the network. If it receives a reply (from a BootP server), it changes its network parameters (IP address, subnet mask and default gateway address) to the values provided. If there is no reply from a BootP server and if DHCP is enabled (DHCPEnable = 1), the gateway initiates a standard DHCP procedure to configure its network parameters.

After changing the network parameters, the gateway attempts to load the *cmp* and various configuration files from the TFTP server's IP address, received from the BootP/DHCP servers. If a TFTP server's IP address isn't received, the gateway attempts to load the software (*cmp*) file and / or configuration files from a preconfigured TFTP server (refer to the parameters 'IniFileURL' and 'CmpFileURL' described in [Table 6-1](#) on page 90). Thus, the gateway can obtain its network parameters from BootP or DHCP servers and its software and configuration files from a different TFTP server (preconfigured in *ini* file).

If BootP/DHCP servers are not found or when the gateway is reset from the Web / SNMP, it retains its network parameters and attempts to load the software (*cmp*) file and / or configuration files from a preconfigured TFTP server.

If a preconfigured TFTP server doesn't exist, the gateway operates using the existing software and configuration files loaded on its non-volatile memory.

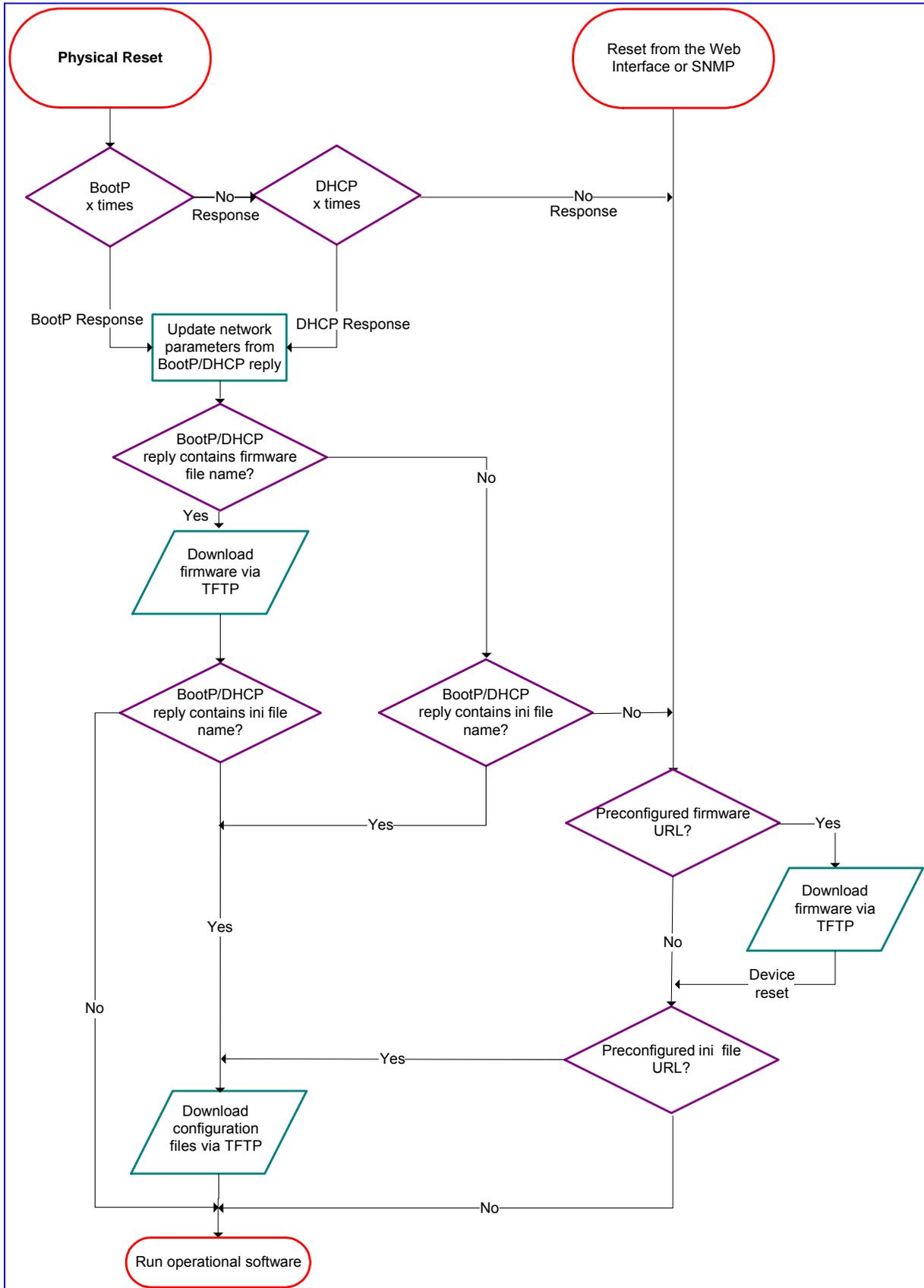
Note that after the operational software runs, if DHCP is configured, the gateway attempts to renew its lease with the DHCP server.



**Note 1:** Though DHCP and BootP servers are very similar in operation, the DHCP server includes some differences that could prevent its operation with BootP clients. However, many DHCP servers, such as Windows™ NT DHCP server, are backward-compatible with BootP protocol and can be used for gateway configuration.

**Note 2:** The time duration between BootP/DHCP requests is set to 1 second by default. This can be changed by the *BootPDelay* *ini* file parameter. Also, the number of requests is 3 by default and can be changed by *BootPRetrieves* *ini* file parameter. (Both parameters can also be set using the BootP command line switches).

Figure 10-1: Mediant 2000 Startup Process



## 10.2 DHCP Support

When the gateway is configured to use DHCP (DHCPEnable = 1), it attempts to contact the enterprise's DHCP server to obtain the networking parameters (IP address, subnet mask, default gateway, primary/secondary DNS server and SIP server address). These network parameters have a "time limit". After the time limit expires, the gateway must "renew" its lease from the DHCP server.

Note that if the DHCP server denies the use of the gateway's current IP address and specifies a different IP address (according to RFC 1541), the gateway must change its networking parameters. If this happens while calls are in progress, they are not automatically rerouted to the new network address (since this function is beyond the scope of a VoIP gateway). Therefore, administrators are advised to configure DHCP servers to allow renewal of IP addresses.

**Note:** If the gateway's network cable is disconnected and reconnected, a DHCP renewal is performed (to verify that the gateway is still connected to the same network).

When DHCP is enabled, the gateway also includes its product name (e.g., 'Mediant 2000') in the DHCP 'option 60' Vendor Class Identifier. The DHCP server can use this product name to assign an IP address accordingly.

**Note:** After power-up, the gateway issues two DHCP requests. Only in the second request, the DHCP 'option 60' is contained. If the gateway is reset from the Web/SNMP, only a single DHCP request containing 'option 60' is sent.

If DHCP procedure is used, the new gateway IP address, allocated by the DHCP server, must be detected.



**Note:** If, during operation, the IP address of the gateway is changed as a result of a DHCP renewal, the gateway is automatically reset.

➤ **To detect the gateway's IP address, follow one of the procedures below:**

- Starting with Bootload software version 1.92, the gateway can use host name in the DHCP request. The host name is set to `acl_nnnnn`, where `nnnnn` stands for the gateway's serial number (the serial number is equal to the last 6 digits of the MAC address converted from Hex to decimal). If the DHCP Server registers this host name to a DNS server, the user can access the gateway (through a Web browser) using a URL of `http://acl_<serial number>` (instead of using the gateway's IP address). For example, if the gateway's MAC address is 00908f010280, the DNS name is `acl_66176`.
- After physically resetting the gateway its IP address is displayed in the 'Client Info' column in the BootP/TFTP configuration utility (refer to [Figure B-1](#) on page 191).
- Contact your System Administrator.

## 10.3 BootP Support

### 10.3.1 Upgrading the Mediant 2000

When upgrading the Mediant 2000 (loading new software onto the gateway) using the BootP/TFTP configuration utility:

- From version 4.2 to version 4.4, the device loses its configuration. Therefore, to retain the previous gateway configuration you must save the *ini* file before you replace the *cmp* file, and reload it to the device. For information on backing up and restoring the gateway's configuration, refer to Section 5.9.5 on page 69.

- From version 4.4 to version 4.4 or to any higher version, the device retains its configuration (*ini* file), however, the auxiliary files (CPT, logo, etc.) may be erased.

When using the Software Upgrade wizard, available through the Web Interface (refer to Section 5.11.1 on page 78), the auxiliary files are saved as well.

**Note:** To save the *cmp* file to non-volatile memory, use the *-fb* command line switches. If the file is not saved, the gateway reverts to the old version of software after the next reset. For information on using command line switches, refer to Section B.11.6 on page 198.

### 10.3.2 Vendor Specific Information Field

The Mediant 2000 uses the vendor specific information field in the BootP request to provide device-related initial startup information. The BootP/TFTP configuration utility displays this information in the 'Client Info' column (refer to Figure B-1).

**Note:** This option is not available on DHCP servers.

The Vendor Specific Information field is disabled by default. To enable / disable this feature: set the *ini* file parameter 'ExtBootPReqEnable' (Table 6-1 on page 90) or use the '*-be*' command line switch (refer to Table B-1 on page 198).

Table 10-1 details the vendor specific information field according to device types:

**Table 10-1: Vendor Specific Information Field**

Tag #	Description	Value	Length
220	Board Type	#02 = IPM-1610/TP-1610 #03 = IPMTP260 #05 = IPMTP260	1
221	Current IP Address	XXX.XXX.XXX.XXX	4
222	Burned Boot Software Version	X.XX	4
223	Burned <i>cmp</i> Software Version	XXXXXXXXXXXX	12
224	Geographical Address	0 – 31 (TP-260 Only)	1
225	Chassis Geographical Address	0 – 31 (TP-260 Only)	1
229	E&M	N/A	1

Table 10-2 exemplifies the structure of the vendor specific information field for a TP-1610 slave module with IP Address 10.2.70.1.

**Table 10-2: Structure of the Vendor Specific Information Field**

Vendor-Specific Information Code	Length Total	Tag Num	Length	Value	Tab Num	Length	Value	Tag Num	Length	Value (1)	Value (2)	Value (3)	Value (4)	Tag End
42	12	220	1	2	225	1	1	221	4	10	2	70	1	255

# 11 SNMP-Based Management

Simple Network Management Protocol (SNMP) is a standard-based network control protocol used to manage elements in a network. The SNMP Manager (usually implemented by a Network Manager (NM) or an Element Manager (EM)) connects to an SNMP Agent (embedded on a remote Network Element (NE)) to perform network element Operation, Administration and Maintenance (OAM).

Both the SNMP Manager and the NE refer to the same database to retrieve information or configure parameters. This database is referred to as the Management Information Base (MIB), and is a set of statistical and control values. Apart from the standard MIBs documented in IETF RFCs, SNMP additionally enables the use of private MIBs, containing a non-standard information set (specific functionality provided by the NE).

Directives, issued by the SNMP Manager to an SNMP Agent, consist of the identifiers of SNMP variables (referred to as MIB object identifiers or MIB variables) along with instructions to either get the value for that identifier, or set the identifier to a new value (configuration). The SNMP Agent can also send unsolicited events towards the EM, called SNMP traps.

The definitions of MIB variables supported by a particular agent are incorporated in descriptor files, written in Abstract Syntax Notation (ASN.1) format, made available to EM client programs so that they can become aware of MIB variables and their use.

The device contains an embedded SNMP Agent supporting both general network MIBs (such as the IP MIB), VoP-specific MIBs (such as RTP) and our proprietary MIBs (acBoard, acGateway, acAlarm and other MIBs), enabling a deeper probe into the inter-working of the device. All supported MIB files are supplied to customers as part of the release.

## 11.1 About SNMP

### 11.1.1 SNMP Message Standard

Four types of SNMP messages are defined:

- Get - A request that returns the value of a named object.
- Get-Next - A request that returns the next name (and value) of the 'next' object supported by a network device given a valid SNMP name.
- Set - A request that sets a named object to a specific value.
- Trap - A message generated asynchronously by network devices. It is an unsolicited message from an agent to the manager.

Each of these message types fulfills a particular requirement of Network Managers:

- Get Request - Specific values can be fetched via the 'get' request to determine the performance and state of the device. Typically, many different values and parameters can be determined via SNMP without the overhead associated with logging into the device, or establishing a Transmission Control Protocol (TCP) connection with the device.
- Get Next Request - Enables the SNMP standard network managers to 'walk' through all SNMP values of a device (via the 'get-next' request) to determine all names and values that an operant device supports. This is accomplished by beginning with the first SNMP object to be fetched, fetching the next name with a 'get-next', and repeating this operation.
- Set Request - The SNMP standard provides a method of effecting an action associated with a device (via the 'set' request) to accomplish activities such as disabling interfaces, disconnecting users, clearing registers, etc. This provides a way of configuring and controlling network devices via SNMP.

- Trap Message - The SNMP standard furnishes a mechanism by which devices can 'reach out' to a Network Manager on their own (via a 'trap' message) to notify or alert the manager of a problem with the device. This typically requires each device on the network to be configured to issue SNMP traps to one or more network devices that are awaiting these traps.

The above message types are all encoded into messages referred to as Protocol Data Units (PDUs) that are interchanged between SNMP devices.

### 11.1.2 SNMP MIB Objects

The SNMP MIB is arranged in a tree-structured fashion, similar in many ways to a disk directory structure of files. The top level SNMP branch begins with the ISO 'internet' directory, which contains four main branches:

- The 'mgmt' SNMP branch - Contains the standard SNMP objects usually supported (at least in part) by all network devices.
- The 'private' SNMP branch - Contains those 'extended' SNMP objects defined by network equipment vendors.
- The 'experimental' and 'directory' SNMP branches - Also defined within the 'internet' root directory, these branches are usually devoid of any meaningful data or objects.

The 'tree' structure described above is an integral part of the SNMP standard, though the most pertinent parts of the tree are the 'leaf' objects of the tree that provide actual management data regarding the device. Generally, SNMP leaf objects can be partitioned into two similar but slightly different types that reflect the organization of the tree structure:

- Discrete MIB Objects - Contain one precise piece of management data. These objects are often distinguished from 'Table' items (below) by adding a '.0' (dot-zero) extension to their names. The operator must merely know the name of the object and no other information.
- Table MIB Objects - Contain multiple sections of management data. These objects are distinguished from 'Discrete' items (above) by requiring a '.' (dot) extension to their names that uniquely distinguishes the particular value being referenced. The '.' (dot) extension is the 'instance' number of an SNMP object. For 'Discrete' objects, this instance number is zero. For 'Table' objects, this instance number is the index into the SNMP table. SNMP tables are special types of SNMP objects which allow parallel arrays of information to be supported. Tables are distinguished from scalar objects, so that tables can grow without bounds. For example, SNMP defines the 'ifDescr' object (as a standard SNMP object) that indicates the text description of each interface supported by a particular device. Since network devices can be configured with more than one interface, this object can only be represented as an array.

By convention, SNMP objects are always grouped in an 'Entry' directory, within an object with a 'Table' suffix. (The 'ifDescr' object described above resides in the 'ifEntry' directory contained in the 'ifTable' directory).

### 11.1.3 SNMP Extensibility Feature

One of the principal components of an SNMP manager is a MIB Compiler which allows new MIB objects to be added to the management system. When a MIB is compiled into an SNMP manager, the manager is made 'aware' of new objects that are supported by agents on the network. The concept is similar to adding a new schema to a database.

Typically, when a MIB is compiled into the system, the manager creates new folders or directories that correspond to the objects. These folders or directories can typically be viewed with a MIB Browser, which is a traditional SNMP management tool incorporated into virtually all Network Management Systems.

The act of compiling the MIB allows the manager to know about the special objects supported by the agent and access these objects as part of the standard object set.

## 11.2 Carrier Grade Alarm System

The basic alarm system has been extended to a carrier-grade alarm system. A carrier-grade alarm system provides a reliable alarm reporting mechanism that takes into account EMS outages, network outages, and transport mechanism such as SNMP over UDP.

A carrier-grade alarm system is characterized by the following:

- The device has a mechanism that allows a manager to determine which alarms are currently active in the device. That is, the device maintains an active alarm table.
- The device has a mechanism to allow a manager to detect lost alarm raise and clear notifications [sequence number in trap, current sequence number MIB object].
- The device has a mechanism to allow a manager to recover lost alarm raise and clear notifications [maintains a log history].
- The device sends a cold start trap to indicate that it is starting. This allows the EMS to synchronize its view of the device's active alarms.

The SNMP alarm traps are sent as in previous releases. This system provides the mechanism for viewing of history and current active alarm information.

### 11.2.1 Active Alarm Table

The device maintains an active alarm table to allow a manager to determine which alarms are currently active in the device. Two views of the active alarm table are supported by the agent:

- `acActiveAlarmTable` in the enterprise `acAlarm`
- `alarmActiveTable` and `alarmActiveVariableTable` in the IETF standard `ALARM-MIB` (rooted in the AC tree)

The `acActiveAlarmTable` is a simple, one-row per alarm table that is easy to view with a MIB browser.

The `ALARM-MIB` is currently a draft standard and therefore has no OID assigned to it. In the current software release, the MIB is rooted in the experimental MIB subtree. In a future release, after the MIB has been ratified and an OID assigned, it is to move to the official OID.

### 11.2.2 Alarm History

The device maintains a history of alarms that have been raised and traps that have been cleared to allow a manager to recover any lost, raised or cleared traps. Two views of the alarm history table are supported by the agent:

- `acAlarmHistoryTable` in the enterprise `acAlarm`
- `nImLogTable` and `nImLogVariableTable` in the standard `NOTIFICATION-LOG-MIB`

As with the `acActiveAlarmTable`, the `acAlarmHistoryTable` is a simple, one-row-per-alarm table that is easy to view with a MIB browser.

## 11.3 Cold Start Trap

Mediant 2000 technology supports a cold start trap to indicate that the device is starting. This allows the manager to synchronize its view of the device's active alarms. Two different traps are sent at start-up:

- The standard coldStart trap - `iso(1).org(3).dod(6).internet(1).snmpV2(6).snmpModules(3).snmpMIB(1).snmpMIBObjects(1).snmpTraps(5).coldStart(1)` - sent at system initialization.
- The enterprise `acBoardEvBoardStarted` which is generated at the end of system initialization. This is more of an 'application-level' cold start sent after the entire initializing process is complete and all the modules are ready.

## 11.4 Third-Party Performance Monitoring Measurements

Performance measurements are available for a third-party performance monitoring system through an SNMP interface. These measurements can be polled at scheduled intervals by an external poller or utility in a media server or other off-device system.

The device provides two types of performance measurements:

1. **Gauges:** Gauges represent the current state of activities on the device. Gauges, unlike counters, can decrease in value, and like counters, can increase. The value of a gauge is the current value or a snapshot of the current activity on the device.
2. **Counters:** Counters always increase in value and are cumulative. Counters, unlike gauges, never decrease in value unless the off-device system is reset. the counters are then zeroed.

Performance measurements are provided by three proprietary MIBs (acPerfMediaGateway, acPerfMediaServices and acPerfH323SIPGateway). The first MIB is a generic-type of performance measurements MIB available on all Mediant 2000 and related devices. The second is specific to the media server, and the third is for H.323/SIP media gateways.

The generic performance measurements MIB covers:

- Control protocol
- RTP stream
- System packets statistics

Performance measurement enterprise MIB supports statistics which apply to the Proxy/Gatekeeper routing tables.

## 11.5 TrunkPack-VoP Series Supported MIBs

The Mediant 2000 contains an embedded SNMP Agent supporting the following MIBs:

- Standard MIB (MIB-II) - The various SNMP values in the standard MIB are defined in RFC 1213. The standard MIB includes various objects to measure and monitor IP activity, TCP activity, UDP activity, IP routes, TCP connections, interfaces and general system indicators.
- RTP MIB - The RTP MIB is supported in conformance with the IETF's RFC 2959. It contains objects relevant to the RTP streams generated and terminated by the device and to RTCP information related to these streams.
- Trunk MIB - The Trunk MIB contains objects relevant to E1/T1 Trunk interfaces.
- NOTIFICATION-LOG-MIB - This standard MIB (RFC 3014 - iso.org.dod.internet.mgmt.mib-2) is supported as part of our implementation of carrier grade alarms.
- ALARM-MIB - This is an IETF proposed MIB also supported as part of our implementation of carrier grade alarms. This MIB is still not standard and is therefore under the audioCodes.acExperimental branch.
- SNMP-TARGET-MIB - This MIB is partially supported (RFC 2273). It allows for the configuration of trap destinations and trusted managers only.
- SNMP Research International Enterprise MIBs - Mediant 2000 supports two SNMP Research International MIBs: SR-COMMUNITY-MIB and TGT-ADDRESS-MASK-MIB. These MIBs are used in the configuration of SNMPv2c community strings and trusted managers.

In addition to the standard MIBs, the complete series contains several proprietary MIBs:

- acBoard MIB - This proprietary MIB contains objects related to configuration of the device and channels, as well as to run-time information. Through this MIB, users can set up the device configuration parameters, reset the device, monitor the device's operational robustness and Quality of Service during run-time, and receive traps.



**Note:** The acBoard MIB is still supported but is being replaced by five newer proprietary MIBs.

The acBoard MIB has the following groups:

- boardConfiguration
- boardInformation
- channelConfiguration
- channelStatus
- reset
- acTrap

As noted above, five new MIBs cover the device's general parameters. Each contains a Configuration subtree for configuring related parameters. In some, there also are Status and Action subtrees.

The 5 MIBs are:

1. AC-ANALOG-MIB
2. AC-CONTROL-MIB
3. AC-MEDIA-MIB
4. AC-PSTN-MIB
5. AC-SYSTEM-MIB

Other proprietary MIBs are:

- acGateway MIB - This proprietary MIB contains objects related to configuration of the device when applied as a SIP or H.323 media gateway only. This MIB complements the other proprietary MIBs.

The acGateway MIB has the following groups:

- **Common** - for parameters common to both SIP and H.323
- **SIP** - for SIP parameters only
- **H.323** - for H.323 parameters only

- acAtm - This proprietary MIB contains objects related to configuration and status of the device when applied as an ATM media gateway only. This MIB complements the other proprietary MIBs.

The acAtm MIB has the following groups:

- acAtmConfiguration - for configuring ATM related parameters
- acAtmStatus - for the status of ATM connections

- acAlarm - This is AudioCodes' proprietary carrier-grade alarm MIB. It is a simpler implementation of the notificationLogMIB and the IETF suggested alarmMIB (both also supported in all AudioCodes' devices).

The acAlarm MIB has the following groups:

- ActiveAlarm - straightforward (single-indexed) table, listing all currently active alarms, together with their bindings (the alarm bindings are defined in acAlarm.acAlarmVarbinds and also in acBoard.acTrap.acBoardTrapDefinitions.oid\_1\_3\_6\_1\_4\_1\_5003\_9\_10\_1\_21\_2\_0).
- acAlarmHistory - straightforward (single-indexed) table, listing all recently raised alarms together with their bindings (the alarm bindings are defined in acAlarm.

acAlarmVarbinds and also in acBoard.acTrap. acBoardTrapDefinitions. oid\_1\_3\_6\_1\_4\_1\_5003\_9\_10\_1\_21\_2\_0).

The table size can be altered via notificationLogMIB.notificationLogMIBObjects.nlmConfig.nlmConfigGlobalEntryLimit or notificationLogMIB.notificationLogMIBObjects.nlmConfig.nlmConfigLogTable.nlmConfigLogEntry.nlmConfigLogEntryLimit.

The table size can be any value between 50 to 1000 and is 500 by default.

- Traps - Full proprietary trap definitions and trap Varbinds are found in the acBoard MIB and acAlarm MIB.

The following proprietary traps are supported in the device:

- acBoardEvResettingBoard - Sent after the device is reset.
- acBoardEvBoardStarted - Sent after the device is successfully restored and initialized following reset.
- acBoardTemperatureAlarm - Sent when a board exceeds its temperature limits.
- acBoardConfigurationError - Sent when a device's settings are illegal - the trap contains a message stating/detailing/explaining the illegality of the setting.
- acBoardFatalError - Sent whenever a fatal device error occurs.
- acFeatureKeyError - Development pending. Intended to relay Feature Key errors, etc.
- acgwAdminStateChange - Sent when Graceful Shutdown commences and ends.
- acBoardCallResourcesAlarm - Indicates that no free channels are available.
- acBoardControllerFailureAlarm - The Gatekeeper/Proxy is not found or registration failed. Internal routing table can be used for routing.
- acBoardEthernetLinkAlarm - Ethernet link or links are down.
- acBoardOverloadAlarm - Overload in one or some of the system's components.
- acActiveAlarmTableOverflow - An active alarm could not be placed in the active alarm table because the table is full.
- acAtmPortAlarm<sup>1</sup> - ATM Port Alarm.
- acAudioProvisioningAlarm<sup>1</sup> - Raised if the Media Server is unable to provision its audio.

In addition to the listed traps, the device also supports the following standard traps:

- dsx1LineStatusChange
- coldStart
- authenticationFailure



**Note 1:** The following are special notes pertaining to MIBs:

- A detailed explanation of each parameter can be viewed in an SNMP browser in the 'MIB Description' field.
- Not all groups in the MIB are functional. Refer to version release notes.
- Certain parameters are non-functional. Their MIB status is marked 'obsolete'.
- When a parameter is set to a new value via SNMP, the change may affect device functionality immediately or may require that the device be soft reset for the change to take effect. This depends on the parameter type.

**Note 2:** The current (updated) device configuration parameters are programmed into the device provided that the user does not load an *ini* file to the device after reset. Loading an *ini* file after reset overrides the updated parameters.

Additional MIBs are to be supported in future releases.

## 11.6 SNMP Interface Details

This section describes details of the SNMP interface that is required when developing an Element Manager (EM) for any of the TrunkPack-VoP Series products, or to manage a device with a MIB browser.

Currently, both SNMP and *ini* file commands and downloads are not encrypted. For *ini* file encoding, refer to Section 6.1 on page 87.

### 11.6.1 SNMP Community Names

By default, the device uses a single, read-only community string of 'public' and a single read-write community string of 'private'.

Users can configure up to 5 read-only community strings and up to 5 read-write community strings, and a single trap community string is supported:

#### 11.6.1.1 Configuration of Community Strings via the *ini* File

```
SNMPREADONLYCOMMUNITYSTRING_<x> = '#####'
```

```
SNMPREADWRITECOMMUNITYSTRING_<x> = '#####'
```

where <x> is a number between 0 and 4, inclusive. Note that the '#' character represents any alphanumeric character. The maximum length of the string is 20 characters.

#### 11.6.1.2 Configuration of Community Strings via SNMP

To configure read-only and read-write community strings, the EM must use the srCommunityMIB. To configure the trap community string, the EM must also use the snmpVacmMIB and the snmpTargetMIB.

➤ **To add a read-only community string (v2user):**

- Add a new row to the srCommunityTable with CommunityName v2user and GroupName ReadGroup.

➤ **To delete the read-only community string (v2user), take these 2 steps:**

1. If v2user is being used as the trap community string, follow the procedure for changing the trap community string (see below).
2. Delete the srCommunityTable row with CommunityName v2user.

➤ **To add a read-write community string (v2admin):**

- Add a new row to the srCommunityTable with CommunityName of v2admin and GroupName ReadWriteGroup.

➤ **To delete the read-write community string (v2admin), take these 2 steps:**

1. If v2admin is being used as the trap community string, follow the procedure for changing the trap community string. (See below.)
2. Delete the srCommunityTable row with a CommunityName of v2admin and GroupName of ReadWriteGroup.

➤ **To change the only read-write community string from v2admin to v2mgr, take these 4 steps:**

1. Follow the procedure above to add a read-write community string to a row for v2mgr.
2. Set up the EM so that subsequent 'set' requests use the new community string, v2mgr.
3. If v2admin is being used as the trap community string, follow the procedure to change the trap community string (see below).
4. Follow the procedure above to delete a read-write community name in the row for v2admin.

➤ **To change the trap community string, take these 2 steps:**

(The following procedure assumes that a row already exists in the srCommunityTable for the new trap community string. The trap community string can be part of the TrapGroup, ReadGroup or ReadWriteGroup. If the trap community string is used solely for sending traps (recommended), it should be made part of the TrapGroup).

1. Add a row to the vacmSecurityToGroupTable with these values: SecurityModel=2, SecurityName=the new trap community string, GroupName=TrapGroup, ReadGroup or ReadWriteGroup. The SecurityModel and SecurityName objects are row indices.



**Note:** You must add GroupName and RowStatus on the same set.

2. Modify the SecurityName field in the sole row of the snmpTargetParamsTable.

## 11.6.2 Trusted Managers

By default, the agent accepts 'get' and 'set' requests from any IP address, as long as the correct community string is used in the request. Security can be enhanced via the use of Trusted Managers. A Trusted Manager is an IP address from which the SNMP Agent accepts and processes 'get' and 'set' requests. An EM can be used to configure up to 5 Trusted Managers.



**Note:** If Trusted Managers are defined, all community strings work from all Trusted Managers. That is, there is no way to associate a community string with particular trusted managers.

### 11.6.2.1 Configuration of Trusted Managers via *ini* File

To set the Trusted Managers table from start-up, write the following in the *ini* file:

```
SNMPTRUSTEDMGR_X = D.D.D.D
```

where X is any integer between 0 and 4 (0 sets the first table entry, 1 sets the second, and so on), and D is an integer between 0 and 255.

### 11.6.2.2 Configuration of Trusted Managers via SNMP

To configure Trusted Managers, the EM must use the srCommunityMIB, the snmpTargetMIB and the TGT-ADDRESS-MASK-MIB.

➤ **To add the first Trusted Manager, take these 3 steps:**

(The following procedure assumes that there is at least one configured read-write community. There are currently no Trusted Managers. The taglist for columns for all srCommunityTable rows are currently empty).

1. Add a row to the `snmpTargetAddrTable` with these values: Name=mgr0, TagList=MGR, Params=v2cparams.
2. Add a row to the `tgtAddressMaskTable` table with these values: Name=mgr0, `tgtAddressMask=255.255.255.255:0`. The agent does not allow creation of a row in this table unless a corresponding row exists in the `snmpTargetAddrTable`.
3. Set the value of the `TransportLabel` field on each non-TrapGroup row in the `srCommunityTable` to MGR.

➤ **To add a subsequent Trusted Manager, take these 2 steps:**

(The following procedure assumes that there is at least one configured read-write community. There are currently one or more Trusted Managers. The taglist for columns for all rows in the `srCommunityTable` are currently set to MGR. This procedure must be performed from one of the existing Trusted Managers).

1. Add a row to the `snmpTargetAddrTable` with these values: Name=mgrN, TagList=MGR, Params=v2cparams, where N is an unused number between 0 and 4.
2. Add a row to the `tgtAddressMaskTable` table with these values: Name=mgrN, `tgtAddressMask=255.255.255.255:0`.

An alternative to the above procedure is to set the `tgtAddressMask` column while you are creating other rows in the table.

➤ **To delete a Trusted Manager (not the final one), take this step:**

(The following procedure assumes that there is at least one configured read-write community. There are currently two or more Trusted Managers. The taglist for columns for all rows in the `srCommunityTable` are currently set to MGR. This procedure must be performed from one of the existing trusted managers, but not the one that is being deleted.

- Remove the appropriate row from the `snmpTargetAddrTable`.

The change takes effect immediately. The deleted trusted manager cannot access the device. The agent automatically removes the row in the `tgtAddressMaskTable`.

➤ **To delete the final Trusted Manager, take these 2 steps:**

(The following procedure assumes that there is at least one configured read-write community. There is currently only one Trusted Manager. The taglist for columns for all rows in the `srCommunityTable` are currently set to MGR. This procedure must be performed from the final Trusted Manager.

1. Set the value of the `TransportLabel` field on each row in the `srCommunityTable` to the empty string.
2. Remove the appropriate row from the `snmpTargetAddrTable`

The change takes effect immediately. All managers can now access the device.

### 11.6.3 SNMP Ports

The SNMP Request Port is 161 and the Trap Port is 162. These ports can be changed by setting parameters in the device *ini* file. The parameter name is:

```
SNMPPort = <port_number>  
Valid UDP port number; default = 161
```

This parameter specifies the port number for SNMP requests and responses. Usually, it should not be specified. Use the default.

## 11.6.4 Multiple SNMP Trap Destinations

An agent can now send traps to up to five managers. For each manager, set the following parameters defined in the `snmpManagersTable` in the `acBoardMIB`:

- `snmpTrapManagerSending`
- `snmpManagerIsUsed`
- `snmpManagerTrapPort`
- `snmpManagerIP`

When `snmpManagerIsUsed` is set to zero (not used), the other three parameters are set to zero.

- `snmpManagerIsUsed` (Default = Disable(0))  
The allowed values are 0 (disable or no) and 1 (enable or yes).
- `snmpManagerIp` (Default = 0.0.0.0)  
This is known as `SNMPMANAGERTABLEIP` in the `ini` file and is the IP address of the manager.
- `SnmpManagerTrapPort` (Default = 162)  
The valid port range for this is 100-4000.
- `snmpManagerTrapSendingEnable` (Default = Enable(1))  
The allowed values are 0 (disable) and 1 (enable).



**Note 1:** Each of these MIB objects is independent and can be set regardless of the state of `snmpManagerIsUsed`.

**Note 2:** If the parameter `IsUsed` is set to 1, the IP address for that row should be supplied in the same SNMP PDU.

### 11.6.4.1 Configuration via the *ini* File

In the Mediant 2000 *ini* file, the parameters below can be set to enable or disable the sending of SNMP traps. Multiple trap destinations can be supported on the device by setting multiple trap destinations in the *ini* file.

`SNMPMANAGERTRAPSENDINGENABLE_<x>` = 0 or 1 indicates if traps are to be sent to the specified SNMP trap manager. A value of '1' means that it is enabled, while a value of '0' means disabled.

`<x>` = a number 0, 1, 2 which is the array element index. Currently, up to 5 SNMP trap managers can be supported.

Figure 11-1 presents an example of entries in a device *ini* file regarding SNMP. The device can be configured to send to multiple trap destinations. The lines in the file below are commented out with the ';' at the beginning of the line. All of the lines below are commented out since the first line character is a semi-colon.

**Figure 11-1: Example of Entries in a Device *ini* file Regarding SNMP**

```

; SNMP trap destinations
; The board maintains a table of trap destinations containing 5 ;rows. The rows are
numbered 0..4. Each block of 4 items below ;apply to a row in the table.

; To configure one of the rows, uncomment all 4 lines in that ;block. Supply an IP
address and if necessary, change the port ;number.
; To delete a trap destination, set ISUSED to 0.
; -change these entries as needed
;SNMPManagerTableIP_0=
;SNMPManagerTrapPort_0=162
;SNMPManagerIsUsed_0=1
;SNMPManagerTrapSendingEnable_0=1
;
;SNMPManagerTableIP_1=
;SNMPManagerTrapPort_1=162
;SNMPManagerIsUsed_1=1
;SNMPManagerTrapSendingEnable_1=1
;
;SNMPManagerTableIP_2=
;SNMPManagerTrapPort_2=162
;SNMPManagerIsUsed_2=1
;SNMPManagerTrapSendingEnable_2=1
;
;SNMPManagerTableIP_3=
;SNMPManagerTrapPort_3=162
;SNMPManagerIsUsed_3=1
;SNMPManagerTrapSendingEnable_3=1
;
;SNMPManagerTableIP_4=
;SNMPManagerTrapPort_4=162
;SNMPManagerIsUsed_4=1
;SNMPManagerTrapSendingEnable_4=1

```



**Note:** The same information configurable in the *ini* file can also be configured via the acBoardMIB.

#### 11.6.4.2 Configuration via SNMP

To configure trap destinations, the EM must use the snmpTargetMIB. Up to 5 trap destinations can be configured.

➤ **To add a trap destination:**

- Add a row to the snmpTargetAddrTable with these values:  
Name=trapN, TagList=AC\_TRAP, Params=v2cparams, where N is an unused number between 0 and 4.

All changes to the trap destination configuration take effect immediately.

➤ **To delete a trap destination:**

- Remove the appropriate row from the snmpTargetAddrTable.

➤ **To modify a trap destination:**

(You can change the IP address and/or port number for an existing trap destination. The same effect can be achieved by removing a row and adding a new row).

- Modify the IP address and/or port number for the appropriate row in the snmpTargetAddrTable.

- **To disable a trap destination:**
  - Change TagList on the appropriate row in the snmpTargetAddrTable to the empty string.
- **To enable a trap destination:**
  - Change TagList on the appropriate row in the snmpTargetAddrTable to 'AC\_TRAP'.

## 11.7 SNMP Manager Backward Compatibility

With support for the Multi Manager Trapping feature, the older acSNMPManagerIP MIB object, synchronized with the first index in the snmpManagers MIB table, is also supported. This is translated in two features:

- SET/GET to either of the two MIB objects is identical.  
i.e., as far as the SET/GET are concerned OID 1.3.6.1.4.1.5003.9.10.1.1.2.7 is identical to OID 1.3.6.1.4.1.5003.9.10.1.1.2.21.1.1.3.
- When setting ANY IP to the acSNMPManagerIP (this is the older parameter, not the table parameter), two more parameters are SET to ENABLE. snmpManagerIsUsed.0 and snmpManagerTrapSendingEnable.0 are both set to 1.

## 11.8 AudioCodes' Element Management System

Using AudioCodes' Element Management System (EMS) is recommended to Customers requiring large deployments (multiple media gateways in globally distributed enterprise offices, for example), that need to be managed by central personnel.

The EMS is not included in the device's supplied package. Contact AudioCodes for detailed information on AudioCodes' EMS and on AudioCodes' EVN - Enterprise VoIP Network – solution for large VoIP deployments.

# 12 Selected Technical Specifications

**Table 12-1: Mediant 2000 Selected Technical Specifications (continues on pages 183 to 185)**

Function	Specification
<b>Trunk &amp; Channel Capacity</b> <sup>1</sup>	
<b>Capacity with E1</b>	1, 2, 4, 8 or 16 E1 spans, 30, 60, 120, 240 or 480 digital channels
<b>Capacity with T1</b>	1, 2, 4, 8 or 16 T1 spans, 24, 48, 96, 192 or 384 digital channels
<b>Voice &amp; Tone Characteristics</b>	
<b>Voice Compression</b>	G.711 PCM at 64 kbps $\mu$ -law/A-law (10, 20, 30, 40, 50, 60, 80, 100, 120 msec) G.723.1 MP-MLQ at 5.3 or 6.3 kbps (30, 60, 90, 120 msec) G.726 at 32 kbps ADPCM (10, 20, 30, 40, 50, 60, 80, 100, 120 msec) G.729 CS-ACELP 8 Kbps Annex A / B (10, 20, 30, 40, 50, 60, 80, 100 msec) NetCoder at 6.4, 7.2, 8.0 and 8.8 kbps (20, 40, 60, 80, 100, 120 msec). EVRC (20, 40, 60, 80, 100 msec). AMR (20 msec only) Transparent (20, 40, 60, 80, 100, 120 msec)
<b>Silence Suppression</b>	G.723.1 Annex A G.729 Annex B PCM and ADPCM: Standard Silence Descriptor (SID) with Proprietary Voice Activity Detection (VAD) and Comfort Noise Generation (CNG) NetCoder
<b>Packet Loss Concealment</b>	G.711 appendix 1 G.723.1 G.729 a/b
<b>Echo Cancellation</b>	G.165 and G.168 2000, configurable tail length per gateway from 32 to 128 msec
<b>DTMF Detection and Generation</b>	Dynamic range 0 to -25 dBm compliant with TIA 464B and Bellcore TR-NWT-000506.
<b>DTMF Transport (in-band)</b>	Mute, transfer in RTP payload or relay in compliance with RFC 2833
<b>Call Progress Tone Detection and Generation</b>	16 tones: single tone or dual tones, programmable frequency & amplitude; 15 frequencies in the range 300 to 1980 Hz, 1 or 2 cadences per tone, up to 2 sets of ON/OFF periods.
<b>Output Gain Control</b>	-32 dB to +31 dB in steps of 1 dB
<b>Input Gain Control</b>	-32 dB to +31 dB in steps of 1 dB
<b>Fax and Modem Transport Modes</b>	
<b>Real time Fax Relay</b>	Group 3 real-time fax relay up to 14400 bps with auto fallback
	Tolerant network delay (up to 9 seconds round trip delay)
	T.30 (PSTN) and T.38 (IP) compliant (real-time fax)
	CNG tone detection & Relay per T.38
	Answer tone (CED or AnsAm) detection & Relay per T.38

<sup>1</sup> Capacity Limitations:

- When the Echo Canceller's length is set to 64 msec or more, the number of available gateway channels is reduced by a factor of 5/6. For detailed information refer to the parameter 'MaxEchoCancellerLength' (Table 6-1).
- When DSP template version 1 (for AMR coder) is selected, the maximum number of channels is 160 instead of 240 (rounded to full E1/T1 trunk capacity 30/24 channels per trunk).
- When DSP template version 2 (for EVRC coder) is selected, the maximum number of channels is 120 instead of 240 (rounded to full E1/T1 trunk capacity 30/24 channels per trunk).

**Table 12-1: Mediant 2000 Selected Technical Specifications (continues on pages 183 to 185)**

Function	Specification
<b>Fax Transparency</b>	Automatic fax bypass (pass-through) to G.711, ADPCM or NSE bypass mode
<b>Modem Transparency</b>	Automatic switching (pass-through) to PCM, ADPCM or NSE bypass mode for modem signals (V.34 or V.90 modem detection)
<b>Protocols</b>	
<b>VoIP Signaling Protocol</b>	SIP - RFC 3261
<b>Communication Protocols</b>	RTP/RTCP packetization. IP stack (UDP, TCP, RTP). Remote Software load (TFTP & BootP support).
<b>Telephony Protocols</b>	PRI (ETSI Euro ISDN, ANSI NI2, 4/5ESS, DMS 100, QSIG Basic Call, Japan INS1500, Australian Telecom, New Zealand Telecom, Hong Kong Variant, Korean MIC) E1/T1 CAS protocols: MFC R2, E&M wink start, Immediate start, delay start, loop start, ground start, Feature Group B, D for E1/T1
<b>In-Band Signaling</b>	DTMF (TIA 464A) MF-R1, MFC R2 User-defined Call Progress Tones
<b>Interfaces</b>	
<b>Telephony Interface</b>	1, 2, 4, 8 or 16 E1/T1/J1 Balanced 120/100 ohm
<b>Network Interface</b>	Two 10/100 Base-TX, half or full duplex with auto-negotiation
<b>LED Indicators</b>	
<b>LED Indications on Front Panel</b>	Power, ACT/Fail, T1/E1 status, LAN status, Swap ready indication
<b>Connectors &amp; Switches</b>	
<b>Rear Panel</b>	
<b>Trunks 1 to 8 and 9 to 16</b>	Two 50-pin female Telco connectors (DDK57AE-40500-21D) or 8 RJ-48c connectors for trunks 1 to 8 only
<b>Ethernet 1 and 2</b>	Two 10/100 Base-TX, RJ-45 shielded connectors
<b>AC Power</b>	Standard IEC320 Appliance inlet. Option for a dual (fully redundant) power supply.
<b>DC Power</b>	2-pin terminal block (screw connection type) suitable for field wiring applications connecting DC Power connector: MSTB2.5/2-STF (5.08 mm) from Phoenix Contact. Bonding and earthing: A 6-32-UNC screw is provided. Correct ring terminal and 16 AWG wire minimum must be used for connection. Or crimp connection shown below.  Note that to meet UL approvals, users <b>must</b> fulfill the criteria below. 2-pin terminal block (crimp connection type) comprising a Phoenix Contact Adaptor: Shroud: MSTBC2,5/2-STZF-5,08. Contacts: MSTBC-MT0,5-1,0 Cable requirement: 18 AWG x 1.5 m length.
<b>Physical</b>	
<b>AC Power Supply</b>	Universal 90 to 260 VAC 1A max, 47-63 Hz Option for a dual redundant power supply.
<b>DC Power Supply (optional)</b>	36 to 72 VDC (nominal 48 VDC), 4A max, floating input
<b>Environmental (DC)</b>	Operation Temp: 0° to 40° C / 32° to 104° F Short Term Operation Temp (per NEBS): 0° to 55° C / 32° to 131° F Storage: -40° to 70° C / -40° to 158° F Humidity: 10 to 90% non-condensing

Table 12-1: Mediant 2000 Selected Technical Specifications (continues on pages 183 to 185)

Function	Specification
<b>Environmental (AC)</b>	Operation Temp: 0° to 40° C / 32° to 104° F Storage: -40° to 70° C / -40° to 158° F Humidity: 10 to 90% non-condensing
<b>Hot Swap</b>	cPCI cards are full hot swap supported Power supplies are redundant but not hot swappable
<b>Enclosure Dimensions</b>	445 x 44 x 300 mm; 17.5 x 1.75 x 12 inch.
<b>Installation</b>	1U 19-inch 2-slot cPCI chassis, Rack mount, shelf or desk top. Rack mount with 2 side brackets, option 2 extra (rear) side brackets.
<b>Type Approvals</b>	
<b>Telecommunication Standards</b>	IC CS03; FCC part 68 Chassis and Host telecom card are approved to the following telecom standards: IC CS03; FCC part 68; CTR 4, CTR 12 & CTR 13; JATE; Anatel, Mexico Telecom, Russia CCC, ASIF S016, ASIF S038.
<b>Safety and EMC Standards</b>	UL 60 950-1, FCC part 15 Class B, (Class A with Sun 2080 CPU card) CE Mark (EN 55022 Class B (Class A with Sun 2080 CPU card), EN 60950-1, EN 55024, EN 300 386), TS001
<b>Environmental</b>	NEBS Level 3: GR-63-Core, GR-1089-Core, Type 1 & 3. Approved for DC powered version. Complies with ETS 301019; ETS 300019-1, -2, -3. (T 1.1, T 2.3, T3.2). Approved for AudioCodes or DC powered versions.
<b>Diagnostics</b>	
<b>Front panel Status LEDs</b>	E1/T1 status LAN status Gateway status (Fail, ACT, Power, and Swap Ready).
<b>Syslog events</b>	Supported by Syslog Server, per RFC 3164 IETF standard.
<b>SNMP MIBs and Traps</b>	SNMP v2c

*All specifications in this document are subject to change without prior notice.*

## Reader's Notes

## Appendix A Mediant 2000 SIP Software Kit

Table A-1 describes the standard supplied software kit for Mediant 2000 SIP gateways. The supplied documentation includes this User's Manual, the Mediant 2000 Fast Track and the Mediant 2000 & TP-1610 SIP Release Notes.

**Table A-1: Mediant 2000 SIP Supplied Software Kit**

File Name	Description
<b>Ram.cmp file</b>	
Mediant_SIP_xxx.cmp	Image file containing the software for the Mediant 2000 gateway.
<b>ini files and utilities</b>	
Mediant_T1.ini	Sample <i>ini</i> file for Mediant 2000 E1 gateways.
Mediant_E1.ini	Sample <i>ini</i> file for Mediant 2000 T1 gateways.
Usa_tones_xx.dat	Default loadable Call Progress Tones <i>dat</i> file.
Usa_tones_xx.ini	Call progress Tones <i>ini</i> file (used to create <i>dat</i> file).
voice_prompts.dat	Default loadable Voice Prompts <i>dat</i> file.
DConvert240.exe	TrunkPack Downloadable Conversion Utility
ACSyslog08.exe	Syslog server.
bootp.exe	BootP/TFTP configuration utility
<b>CAS Protocol Files</b>	Used for various signaling types, such as <i>E_M_WinkTable.dat</i> .
<b>MIB Files</b>	MIB library for SNMP browser
<b>CAS Capture Tool</b>	Utility that is used to convert CAS traces to textual form.
<b>ISDN Capture Tool</b>	Utility that is used to convert ISDN traces to textual form.

## Reader's Notes

## Appendix B The BootP/TFTP Configuration Utility

The BootP/TFTP utility enables you to easily configure and provision our boards and media gateways. Similar to third-party BootP/TFTP utilities (which are also supported) but with added functionality; our BootP/TFTP utility can be installed on Windows™ 98 or Windows™ NT/2000/XP. The BootP/TFTP utility enables remote reset of the device to trigger the initialization procedure (BootP and TFTP). It contains BootP and TFTP utilities with specific adaptations to our requirements.

### B.1 When to Use the BootP/TFTP

The BootP/TFTP utility can be used with the device as an alternative means of initializing the gateways. Initialization provides a gateway with an IP address, subnet mask, and the default gateway IP address. The tool also loads default software, *ini* and other configuration files. BootP Tool can also be used to restore a gateway to its initial configuration, such as in the following instances:

- The IP address of the gateway is not known.
- The Web browser has been inadvertently turned off.
- The Web browser password has been forgotten.
- The gateway has encountered a fault that cannot be recovered using the Web browser.



**Tip:** The BootP is normally used to configure the device's initial parameters. Once this information has been provided, the BootP is no longer needed. All parameters are stored in non-volatile memory and used when the BootP is not accessible.

### B.2 An Overview of BootP

BootP is a protocol defined in RFC 951 and RFC 1542 that enables an internet device to discover its own IP address and the IP address of a BootP on the network, and to obtain the files from that utility that need to be loaded into the device to function.

A device that uses BootP when it powers up broadcasts a BootRequest message on the network. A BootP on the network receives this message and generates a BootReply. The BootReply indicates the IP address that should be used by the device and specifies an IP address from which the unit may load configuration files using Trivial File Transfer Protocol (TFTP) described in RFC 906 and RFC 1350.

### B.3 Key Features

- Internal BootP supporting hundreds of entities.
- Internal TFTP.
- Contains all required data for our products in predefined format.
- Provides a TFTP address, enabling network separation of TFTP and BootP utilities.
- Tools to backup and restore the local database.
- Templates.
- User-defined names for each entity.
- Option for changing MAC address.

- Protection against entering faulty information.
- Remote reset.
- Unicast BootP response.
- User-initiated BootP respond, for remote provisioning over WAN.
- Filtered display of BootP requests.
- Location of other BootP utilities that contain the same MAC entity.
- Common log window for both BootP and TFTP sessions.
- Works with Windows™ 98, Windows™ NT, Windows™ 2000 and Windows™ XP.

## B.4 Specifications

- BootP standards: RFC 951 and RFC 1542
- TFTP standards: RFC 1350 and RFC 906
- Operating System: Windows™ 98, Windows™ NT, Windows™ 2000 and Windows™ XP
- Max number of MAC entries: 200

## B.5 Installation

### ➤ To install the BootP/TFTP on your computer, take these 2 steps:

1. Locate the BootP folder on the VoIP gateway supplied CD ROM and open the file Setup.exe.
2. Follow the prompts from the installation wizard to complete the installation.

### ➤ To open the BootP/TFTP, take these 2 steps:

1. From the **Start** menu on your computer, navigate to **Programs** and then click on **BootP**.
2. The first time that you run the BootP/TFTP, the program prompts you to set the user preferences. Refer to the Section B.10 on page 193 for information on setting the preferences.

## B.6 Loading the *cmp* File, Booting the Device

Once the application is running, and the preferences were set (refer to Section B.10), for each unit that is to be supported, enter parameters into the tool to set up the network configuration information and initialization file names. Each unit is identified by a MAC address. For information on how to configure (add, delete and edit) units, refer to Section B.11 on page 195.

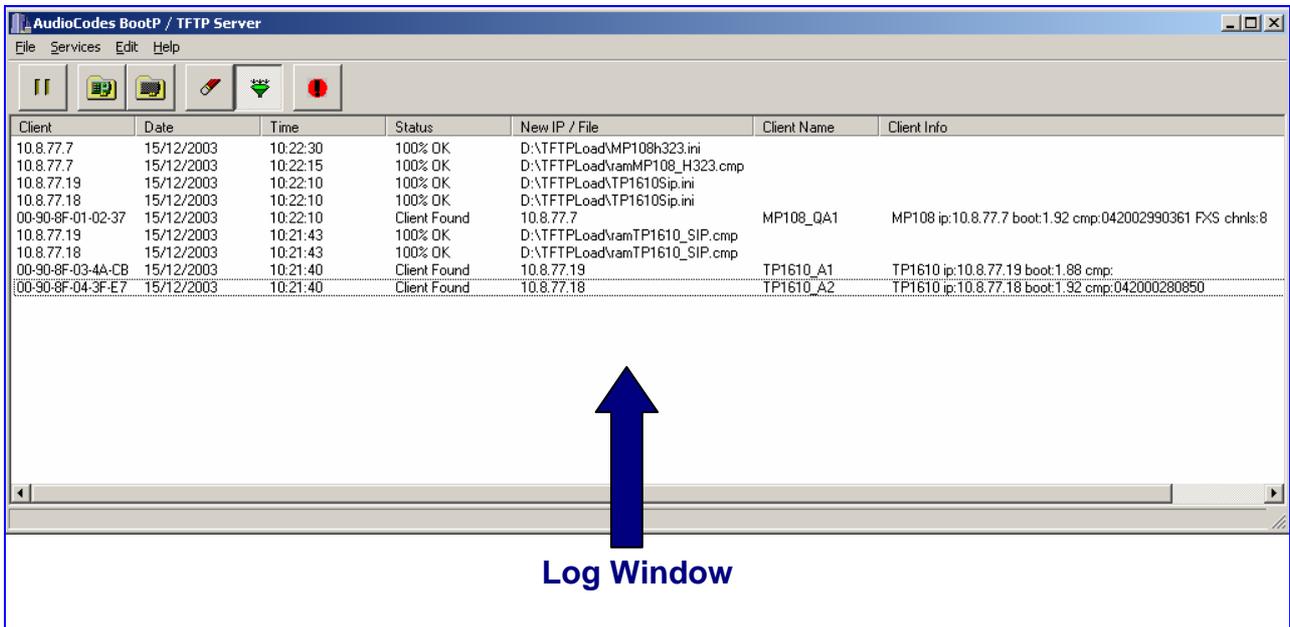
### ➤ To load the software and configuration files, take these 4 steps:

1. Create a folder on your computer that contains all software and configuration files that are needed as part of the TFTP process.
2. Set the BootP and TFTP preferences (refer to Section B.10).
3. Add client configuration for the VoIP gateway that you want to initialize by the BootP, refer to Section B.11.1.
4. Reset the VoIP gateway, either physically or remotely, causing the device to use BootP to access the network and configuration information.

## B.7 BootP/TFTP Application User Interface

Figure B-1 shows the main application screen for the BootP/TFTP utility.

Figure B-1: Main Screen



## B.8 Function Buttons on the Main Screen



**Pause:** Click this button to pause the BootP Tool so that no replies are sent to BootP requests. Click the button again to restart the BootP Tool so that it responds to all BootP requests. The **Pause** button provides a depressed graphic when the feature is active.



**Edit Clients:** Click this button to open a new window that enables you to enter configuration information for each supported VoIP gateway. Details on the Clients window are provided in Section B.11 on page 195.



**Edit Templates:** Click this button to open a new window that enables you to create or edit standard templates. These templates can be used when configuring new clients that share most of the same settings. Details on the **Templates** window are provided in Section B.12 on page 199.



**Clear Log:** Click this button to clear all entries from the Log Window portion of the main application screen. Details on the log window are provided in Section B.9 on page 192.

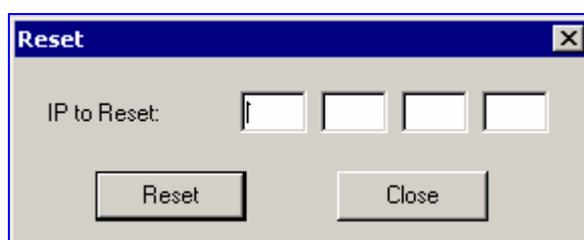


**Filter Clients:** Click this button to prevent the BootP Tool from logging BootP requests received from disabled clients or from clients which do not have entries in the Clients table.



**Reset:** Click this button to open a new window where you enter an IP address requests for a gateway that you want to reset. Refer to Figure B-2 below.

Figure B-2: Reset Screen



When a gateway resets, it first sends a BootRequest. Therefore, Reset can be used to force a BootP session with a gateway without needing to power cycle the gateway. As with any BootP session, the computer running the BootP Tool must be located on the same subnet as the controlled VoIP gateway.

## B.9 Log Window

The log window (refer to [Figure B-1](#) on the previous page) records all BootP request and BootP reply transactions, as well as TFTP transactions. For each transaction, the log window displays the following information:

- **Client:** shows the Client address of the VoIP gateway, which is the MAC address of the client for BootP transactions or the IP address of the client for TFTP transactions.
- **Date:** shows the date of the transaction, based on the internal calendar of the computer.
- **Time:** shows the time of day of the transaction, based on the internal clock of the computer.
- **Status:** indicates the status of the transaction.
  - *Client Not Found:* A BootRequest was received but there is no matching client entry in the BootP Tool.
  - *Client Found:* A BootRequest was received and there is a matching client entry in the BootP Tool. A BootReply is sent.
  - *Client's MAC Changed:* There is a client entered for this IP address but with a different MAC address.
  - *Client Disabled:* A BootRequest was received and there is a matching client entry in the BootP tool but this entry is disabled.
  - *Listed At:* Another BootP utility is listed as supporting a particular client when the Test Selected Client button is clicked (for details on Testing a client, refer to [Section B.11.4](#) on page 196).
  - *Download Status:* Progress of a TFTP load to a client, shown in %.
- **New IP / File:** shows the IP address applied to the client as a result of the BootP transaction, as well as the file name and path of a file transfer for a TFTP transaction.
- **Client Name:** shows the client name, as configured for that client in the Client Configuration screen.

Use right-click on a line in the Log Window to open a pop-up window with the following options:

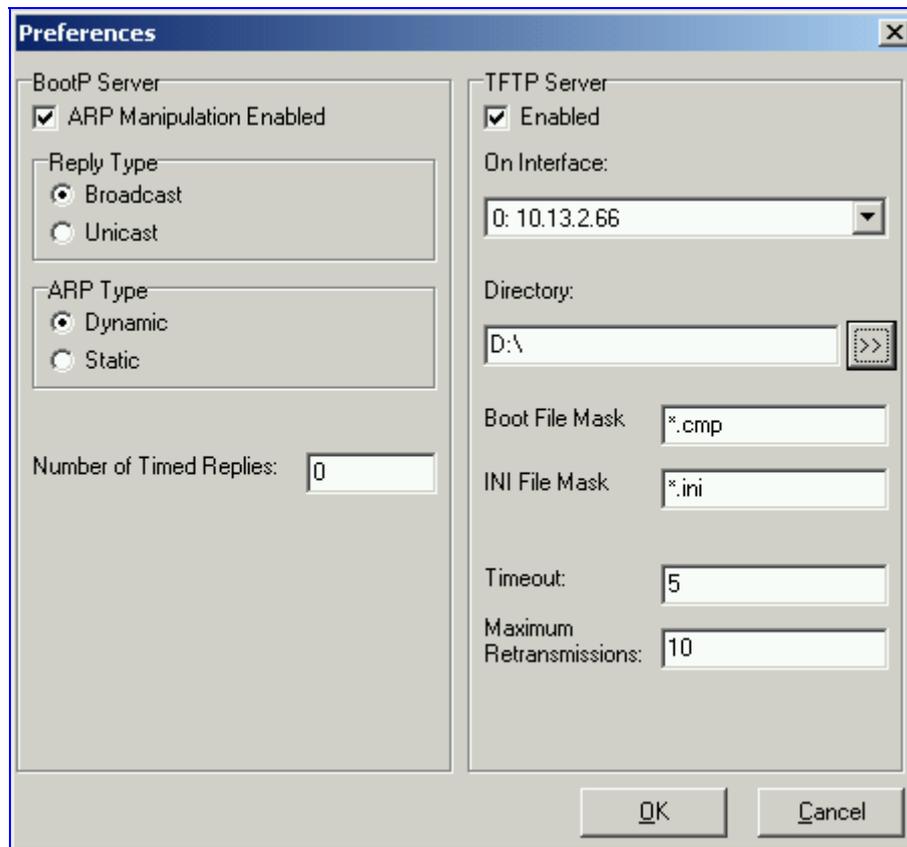
- **Reset:** Selecting this option results in a reset command being sent to the client VoIP gateway. The program searches its database for the MAC address indicated in the line. If the client is found in that database, the program adds the client MAC address to the Address Resolution Protocol (ARP) table for the computer. The program then sends a reset command to the client. This enables a reset to be sent without knowing the current IP address of the client, as long as the computer sending the reset is on the same subnet.
 

**Note:** In order to use reset as described above, the user must have administrator privileges on the computer. Attempting to perform this type of reset without administrator privileges on the computer results in an error message. **ARP Manipulation Enable** must also be turned on in the **Preferences** window.
- **View Client:** Selecting this option, or double clicking on the line in the log window, opens the **Client Configuration** window. If the MAC address indicated on the line exists in the client database, it is highlighted. If the address is not in the client database, a new client is added with the MAC address filled out. You can enter data in the remaining fields to create a new client entry for that client.

## B.10 Setting the Preferences

The Preferences window, [Figure B-3](#), is used to configure the BootP Tool parameters.

**Figure B-3: Preferences Screen**



### B.10.1 BootP Preferences

ARP is a common acronym for Address Resolution Protocol, and is the method used by all Internet devices to determine the link layer address, such as the Ethernet MAC address, in order to route Datagrams to devices that are on the same subnet.

When ARP Manipulation is enabled on this screen, the BootP Tool creates an ARP cache entry on your computer when it receives a BootP BootRequest from the VoIP gateway. Your computer uses this information to send messages to the VoIP gateway without using ARP again. This is particularly useful when the gateway does not yet have an IP address and, therefore, cannot respond to an ARP.

Because this feature creates an entry in the computer ARP cache, Administrator Privileges are required. If the computer is not set to allow administrator privileges, ARP Manipulation cannot be enabled.

- **ARP Manipulation Enabled:** Enable ARP Manipulation to remotely reset a gateway that does not yet have a valid IP address.

If ARP Manipulation is enabled, the following two commands are available.

- **Reply Type:** Reply to a BootRequest can be either **Broadcast** or **Unicast**. The default for the BootP Tool is **Broadcast**. In order for the reply to be set to **Unicast**, ARP Manipulation must first be enabled. This then enables the BootP Tool to find the MAC address for the client in the ARP cache so that it can send a message directly to the requesting device. Normally, this setting can be left at **Broadcast**.

- **ARP Type:** The type of entry made into the ARP cache on the computer, once **ARP Manipulation** is enabled, can be either **Dynamic** or **Static**. Dynamic entries expire after a period of time, keeping the cache clean so that stale entries do not consume computer resources. The Dynamic setting is the default setting and the setting most often used. Static entries do not expire.
- **Number of Timed Replies:** This feature is useful for communicating to VoIP gateways that are located behind a firewall that would block their BootRequest messages from getting through to the computer that is running the BootP Tool. You can set this value to any whole digit. Once set, the BootP Tool can send that number of BootReply messages to the destination immediately after you send a remote reset to a VoIP gateway at a valid IP address. This enables the replies to get through to the VoIP gateway even if the BootRequest is blocked by the firewall. To turn off this feature, set the **Number of Timed Replies = 0**.

## B.10.2 TFTP Preferences

- **Enabled:** To enable the TFTP functionality of the BootP Tool, check the box beside this heading. If you want to use another TFTP application, other than the one included with the BootP Tool, unselect the box.
- **On Interface:** This pull down menu displays all network interfaces currently available on the computer. Select the interface that you want to use for the TFTP. Normally, there is only one choice.
- **Directory:** This option is enabled only when the TFTP is enabled. Use this parameter to specify the folder that contains the files for the TFTP utility to manage (*cmp*, *ini*, Call Progress Tones, etc.).
- **Boot File Mask:** Boot File Mask specifies the file extension used by the TFTP utility for the boot file that is included in the BootReply message. This is the file that contains VoIP gateway software and normally appears as *cmp*.
- **ini File Mask:** *ini* File mask specifies the file extension used by the TFTP utility for the configuration file that is included in the BootReply message. This is the file that contains VoIP gateway configuration parameters and normally appears as *ini*.
- **Timeout:** This specifies the number of seconds that the TFTP utility waits before retransmitting TFTP messages. This can be left at the default value of 5 (the more congested your network, the higher the value you should define in these fields).
- **Maximum Retransmissions:** This specifies the number of times that the TFTP utility tries to resend messages after timing out. This can be left at the default value of 10 (the more congested your network, the higher the value you should define in these fields).

## B.11 Configuring the BootP Clients

The Clients window, shown in [Figure B-4](#) below, is used to set up the parameters for each specific VoIP gateway.

Figure B-4: Client Configuration Screen

MAC	Name	IP
00-90-8F-10-22-33		10.8.201.120
00-90-8F-55-42-21		10.8.201.1
00-90-8F-64-64-12		10.8.201.10

Client MAC: 00-90-8F-64-64-12

Client Name:

Template: <none>

IP: 10 8 201 10

Subnet: 255 255 0 0

Gateway: 10 8 0 1

TFTP Server IP: 10 8 1 21

Boot File: xxx.cmp

INI File: xxx.ini

Buttons: OK, Apply, Apply & Reset

### B.11.1 Adding Clients

Adding a client creates an entry in the BootP Tool for a specific gateway.

➤ **To add a client to the list without using a template, take these 3 steps:**

1. Click on the **Add New Client** icon; a client with blank parameters is displayed. 
2. Enter values in the fields on the right side of the window, using the guidelines for the fields in [Section B.11.5](#) on page 197.
3. Click **Apply** to save this entry to the list of clients, or click **Apply & Reset** to save this entry to the list of clients and send a reset message to that gateway to immediately implement the settings.

**Note:** To use **Apply & Reset** you must enable **ARP Manipulation** in the **Preferences** window. Also, you must have administrator privileges for the computer you are using.

An easy way to create several clients that use similar settings is to create a template. For information on how to create a template, refer to [Section B.12](#) on page 199.

➤ **To add a client to the list using a template, take these 5 steps:**

1. Click on the **Add New Client** icon; a client with blank parameters is displayed. 
2. In the field **Template**, located on the right side of the **Client Configuration Window**, click on the down arrow to the right of the entry field and select the template that you want to use.
3. The values provided by the template are automatically entered into the parameter fields on the right side of the **Client Configuration Window**. To use the template parameters, leave the check box next to that parameter selected. The parameter values appear in gray text.
4. To change a parameter to a different value, unselect the check box to the right of that parameter. This clears the parameter provided by the template and enables you to edit the entry. Clicking the check box again restores the template settings.
5. Click **Apply** to save this entry to the list of clients or click **Apply & Reset** to save this entry to the list of clients and send a reset message to that gateway to immediately implement the settings.  
**Note:** To use **Apply & Reset** you must enable **ARP Manipulation** in the **Preferences** window. Also, you must have administrator privileges for the computer you are using.

## B.11.2 Deleting Clients

➤ **To delete a client from the BootP Tool, take these 3 steps:**

1. Select the client that you wish to delete by clicking on the line in the window for that client.
2. Click the **Delete Current Client** button 
3. A warning pops up. To delete the client, click **Yes**.

## B.11.3 Editing Client Parameters

➤ **To edit the parameters for an existing client, take these 4 steps:**

1. Select the client that you wish to edit by clicking on the line in the window for that client.
2. Parameters for that client display in the parameter fields on the right side of the window.
3. Make the changes required for each parameter.
4. Click **Apply** to save the changes, or click **Apply & Reset** to save the changes and send a reset message to that gateway to immediately implement the settings.  
**Note:** To use **Apply & Reset** you must enable **ARP Manipulation** in the **Preferences** window. Also, you must have administrator privileges for the computer you are using.

## B.11.4 Testing the Client

There should only be one BootP utility supporting any particular client MAC active on the network at any time.

➤ **To check if other BootP utilities support this client, take these 4 steps:**

1. Select the client that you wish to test by clicking on the client name in the main area of the **Client Configuration Window**.
2. Click the Test Selected Client button 
3. Examine the Log Window on the Main Application Screen. If there is another BootP utility that supports this client MAC, there is a response indicated from that utility showing the status Listed At along with the IP address of that utility.
4. If there is another utility responding to this client, you must remove that client from either this utility or the other one.

## B.11.5 Setting Client Parameters

Client parameters are listed on the right side of the **Client Configuration Window**.

- **Client MAC:** The Client MAC is used by BootP to identify the VoIP gateway. The MAC address for the VoIP gateway is printed on a label located on the VoIP gateway hardware. Enter the Ethernet MAC address for the VoIP gateway in this field. Click the box to the right of this field to enable this particular client in the BootP tool (if the client is disabled, no replies are sent to BootP requests).  
**Note:** When the MAC address of an existing client is edited, a new client is added, with the same parameters as the previous client.
- **Client Name:** Enter a descriptive name for this client so that it is easier to remember which VoIP gateway the record refers to. For example, this name could refer to the location of the gateway.
- **Template:** Click the pull down arrow if you wish to use one of the templates that you configured. This applies the parameters from that template to the remaining fields. Parameter values that are applied by the template are indicated by a check mark in the box to the right of that parameter. Uncheck this box if you want to enter a different value. If templates are not used, the box to the right of the parameters is colored gray and is not selectable.
- **IP:** Enter the IP address you want to apply to the VoIP gateway. Use the normal dotted decimal format.
- **Subnet:** Enter the subnet mask you want to apply to the VoIP gateway. Use the normal dotted decimal format. Ensure that the subnet mask is correct. If the address is incorrect, the VoIP gateway may not function until the entry is corrected and a BootP reset is applied.
- **Gateway:** Enter the IP address for the data network gateway used on this subnet that you want to apply to the VoIP gateway. The data network gateway is a device, such as a router, that is used in the data network to interface this subnet to the rest of the enterprise network.
- **TFTP Server IP:** This field contains the IP address of the TFTP utility that is used for file transfer of software and initialization files to the gateway. When creating a new client, this field is populated with the IP address used by the BootP Tool. If a different TFTP utility is to be used, change the IP address in this field to the IP address used by the other utility.
- **Boot File:** This field specifies the file name for the software (*cmp*) file that is loaded by the TFTP utility to the VoIP gateway after the VoIP gateway receives the BootReply message. The actual software file is located in the TFTP utility directory that is specified in the BootP **Preferences** window. The software file can be followed by command line switches. For information on available command line switches, refer to Section [B.11.6](#) on page [198](#).



**Note 1:** Once the software file loads into the gateway, the gateway begins functioning from that software. In order to save this software to non-volatile memory, (only the *cmp* file, i.e., the compressed firmware file, can be burned to your device's flash memory), the `-fb` flag must be added to the end of the file name. If the file is not saved, the gateway reverts to the old version of software after the next reset.

**Note 2:** The **Boot file** field can contain up to two file names: *cmp* file name to be used for load of application image and *ini* file name to be used for gateway provisioning. Either one, two or no file names can appear in the **Boot file** field. To use both file names use the ";" separator (without blank spaces) between the *xxx.cmp* and the *yyy.ini* files (e.g., *ram.cmp;SIPgw.ini*).

- **ini File:** This field specifies the configuration *ini* file that the gateway uses to program its various settings. Enter the name of the file that is loaded by the TFTP utility to the VoIP gateway after it receives the BootReply message. The actual *ini* file is located in the TFTP utility directory that is specified in the BootP Preferences window.

- **Call Agent:** This field specifies the IP address of the MGCP Call Agent that is controlling the gateway. This field can be ignored for all other control/signaling protocols.

## B.11.6 Using Command Line Switches

You can add command line switches in the field **Boot File**.

### ➤ To use a Command Line Switch, take these 4 steps:

1. In the field **Boot File**, leave the file name defined in the field as it is (e.g., *ramxxx.cmp*).
2. Place your cursor after *cmp*
3. Press the space bar
4. Type in the switch you require.

Example: “*ramxxx.cmp -fb*” to burn flash memory.

“*ramxxx.cmp -fb -em 4*” to burn flash memory and for Ethernet Mode 4 (auto-negotiate).

Table B-1 lists and describes the switches that are available:

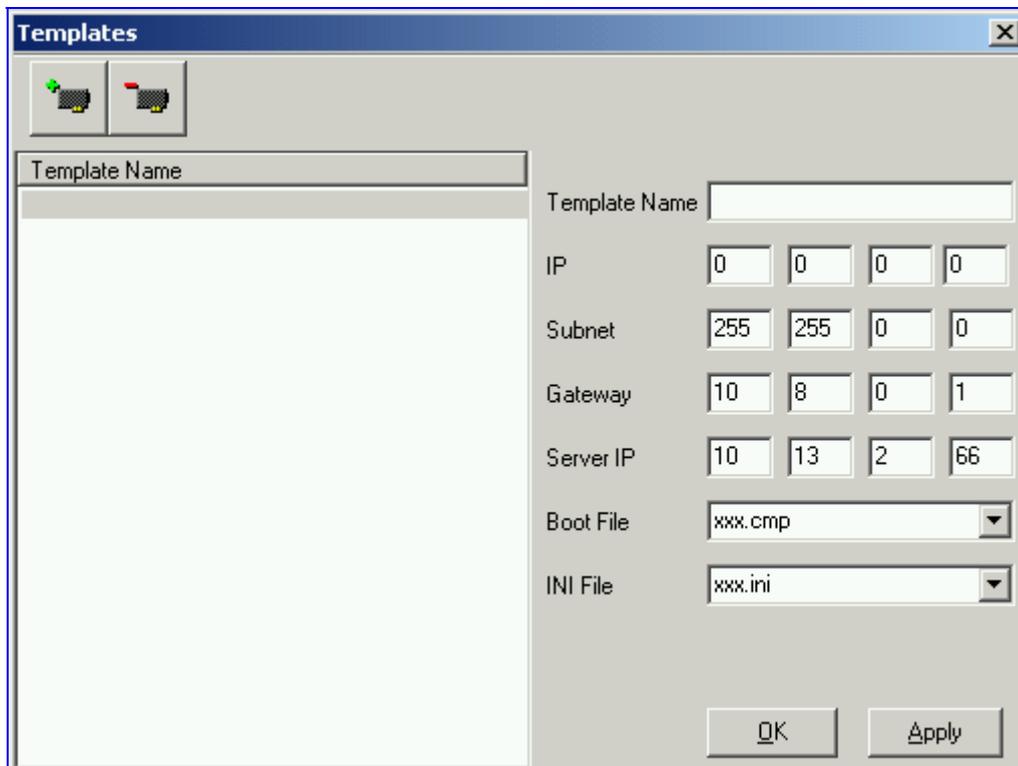
**Table B-1: Command Line Switch Descriptions**

Switch	Description
-fb	Burn <i>ram.cmp</i> in flash (only for <i>cmp</i> files)
-em #	Use this switch to set Ethernet mode. 0 = 10 Base-T half-duplex 1 = 10 Base-T full-duplex 2 = 100 Base-TX half-duplex 3 = 100 Base-TX full-duplex 4 = auto-negotiate (default) Auto-negotiate falls back to half-duplex mode when the opposite port is not in auto-negotiate but the speed (10 Base-T or 100 Base-TX) in this mode is always configured correctly.
-br	BootP retries. Sets the number of BootP requests the device sends during start-up. The device stops sending BootP requests when either BootP reply is received or Number of Retries is reached. This switch takes effect only from the next device reset. 1 = 1 BootP retry, 1 second 2 = 2 BootP retries, 3 seconds 3 = 3 BootP retries, 6 seconds 4 = 10 BootP retries, 30 seconds 5 = 20 BootP retries, 60 seconds 6 = 40 BootP retries, 120 seconds 7 = 100 BootP retries, 300 seconds 15 = BootP retries indefinitely
-bd	BootP delays. Sets the interval between the device's start-up and the first BootP/DHCP request that is issued by the device. The switch only takes effect from the next reset of the device. 1 = 1 second delay (default). 2 = 10 second delay. 3 = 30 second delay. 4 = 60 second delay. 5 = 120 second delay.
-bs	Use <code>-bs 1</code> to enable the Selective BootP mechanism. Use <code>-bs 0</code> to disable the Selective BootP mechanism. The Selective BootP mechanism enables the gateway's integral BootP client to filter unsolicited BootP/DHCP replies (accepts only BootP replies that contain the text "AUDC" in the vendor specific information field). This option is useful in environments where enterprise BootP/DHCP servers provide undesired responses to the gateway's BootP requests.
-be	Use <code>-be 1</code> for the device to send device-related initial startup information (such as board type, current IP address, software version, etc.) in the vendor specific information field (in the BootP request). This information can be viewed in the main screen of the BootP/TFTP, under column 'Client Info' (refer to <a href="#">Figure B-1</a> showing BootP/TFTP main screen with the column 'Client Info' on the extreme right). For a full list of the vendor specific Information fields, refer to <a href="#">Section 10.3</a> on page 169. <b>Note:</b> This option is not available on DHCP servers.

## B.12 Managing Client Templates

Templates can be used to simplify configuration of clients when most of the parameters are the same.

Figure B-5: Templates Screen



➤ **To create a new template, take these 4 steps:**

1. Click on the **Add New Template** button 
2. Fill in the default parameter values in the parameter fields.
3. Click **Apply** to save this new template.
4. Click **OK** when you are finished adding templates.

➤ **To edit an existing template, take these 4 steps:**

1. Select the template by clicking on its name from the list of templates in the window.
2. Make changes to the parameters, as required.
3. Click **Apply** to save this new template.
4. Click **OK** when you are finished editing templates.

➤ **To delete an existing template, take these 3 steps:**

1. Select the template by clicking its name from the list of templates in the window.
2. Click on the **Delete Current Template** button. 
3. A warning pop up message appears. To delete the template, click **Yes**. Note that if this template is currently in use, the template cannot be deleted.

## Reader's Notes

## Appendix C RTP/RTCP Payload Types and Port Allocation

RTP Payload Types are defined in RFC 1889 and RFC 1890. We have added new payload types to enable advanced use of other coder types. These types are reportedly not used by other applications.



**Note:** Refer to the Mediant 2000 & TP-1620 SIP Release Notes for the supported coders.

### C.1 Payload Types Defined in RFC 1890

Table C-1: Packet Types Defined in RFC 1890

Payload Type	Description	Basic Packet Rate [msec]
0	G.711 $\mu$ -Law	10,20
2	G.726-32	10,20
4	G.723 (6.3/5.3 kbps)	30
8	G.711 A-Law	10,20
18	G.729A	20
200	RTCP Sender Report	Randomly, approximately every 5 seconds (when packets are sent by channel)
201	RTCP Receiver Report	Randomly, approximately every 5 seconds (when channel is only receiving)
202	RTCP SDES packet	
203	RTCP BYE packet	
204	RTCP APP packet	

### C.2 Defined Payload Types

Table C-2: Defined Payload Types (continues on pages 201 to 202)

Payload Type	Description	Basic Packet Rate [msec]
35	G.726 32 kbps	20
36	G.726 24 kbps	20
38	G.726 40 kbps	20
39	G.727 16 kbps	20
40	G.727 24-16 kbps	20
41	G.727 24 kbps	20
42	G.727 32-16 kbps	20
43	G.727 32-24 kbps	20
44	G.727-32 kbps	20
45	G.727 40-16 kbps	20
46	G.727 40-24 kbps	20
47	G.727 40-32 kbps	20
49	NetCoder 4.8 kbps	20
50	NetCoder 5.6 kbps	20
51	NetCoder 6.4 kbps	20
52	NetCoder 7.2 kbps	20

Payload Type	Description	Basic Packet Rate [msec]
53	NetCoder 8.0 kbps	20
54	NetCoder 8.8 kbps	20
55	NetCoder 9.6 kbps	20
56	Transparent PCM	20
96	DTMF relay per RFC 2833	
102	Fax Bypass	20
103	Modem Bypass	20
104	RFC 2198 (Redundancy)	Same as channel's voice coder.
105	NSE Bypass	

### C.3 Default RTP/RTCP/T.38 Port Allocation

The following table describes Mediant 2000 gateway default RTP/RTCP/T.38 Port Allocation.

**Table C-3: Default RTP/RTCP/T.38 Port Allocation**

Channel Number	RTP Port	RTCP Port	T.38 Port
1	6000	6001	6002
2	6010	6011	6012
3	6020	6021	6022
4	6030	6031	6032
5	6040	6041	6042
6	6050	6051	6052
7	6060	6061	6062
8	6070	6071	6072
:	:	:	:
<b>n</b>	<b>6000 + 10(n-1)</b>	<b>6001 + 10(n-1)</b>	<b>6002 + 10(n-1)</b>
:	:	:	:
96	6950	6951	6952
:	:	:	:
120	7190	7191	7192
:	:	:	:
192	7910	7911	7912
:	:	:	:
240	8390	8391	8392
:	:	:	:
384	9830	9831	9832
:	:	:	:
480	10790	10791	10792



**Note:** To configure the gateway to use the same port for both RTP and T.38 packets, set the parameter 'T38UseRTPPort' to 1.

## Appendix D Fax and Modem Transport Modes

### D.1 Fax/Modem Settings

Users can choose to use for fax, and for each modem type (V.22/V.23/Bell/V.32/V.34), one of the following transport methods:

- Fax relay mode (demodulation / remodulation, not applicable to Modem),
- Bypass (using a high bit rate coder to pass the signal), or
- Transparent (passing the signal in the current voice coder).

When any of the relay modes are enabled, distinction between fax and modem is not immediately possible at the beginning of a session. The channel is therefore in "Answer Tone" mode until a decision is made. The packets sent to the network at this stage are T.38-complaint fax relay packets.

#### D.1.1 Configuring Fax Relay Mode

When FaxTransportType= 1 (relay mode), then when fax is detected the channel automatically switches from the current voice coder to answer tone mode, and then to T.38-complaint fax relay mode.

When fax transmission ends, the reverse is carried out, and fax relay switches to voice. This mode switch occurs automatically, both at the local and remote endpoints.

Users can limit the fax rate using the FaxRelayMaxRate parameter and can enable/disable ECM fax mode using the FaxRelayECMEnable parameter.

When using T.38 mode, the User can define a redundancy feature to improve fax transmission over congested IP network. This feature is activated by "FaxRelayRedundancyDepth" and "EnhancedFaxRelayRedundancyDepth" parameters. Although this is a proprietary redundancy scheme, it should not create problems when working with other T.38 decoders.



**Note:** T.38 mode currently supports only the T.38 UDP syntax.

#### D.1.2 Configuring Fax/Modem ByPass Mode

When VxxTransportType=2 (FaxModemBypass, Vxx can be either V32/V22/Bell/V34/Fax), then when fax/modem is detected, the channel automatically switches from the current voice coder to a high bit-rate coder, as defined by the User, with the FaxModemBypassCoderType configuration parameter.

During the bypass period, the coder uses the packing factor (by which a number of basic coder frames are combined together in the outgoing WAN packet) set by the User in the FaxModemBypassM configuration parameter. The network packets to be generated and received during the bypass period are regular voice RTP packets (per the selected bypass coder) but with a different RTP Payload type.

When fax/modem transmission ends, the reverse is carried out, and bypass coder is switched to regular voice coder.

### D.1.3 Supporting V.34 Faxes

V.34 fax machine support is available only in bypass mode (fax relay is not supported) when the channel is configured in one of the configurations described below:

```
FaxTransportMode = 2 (Bypass)
V34ModemTransportType = 2 (Modem bypass)
```

In this configuration, both T.30 and V.34 faxes work in Bypass mode

**Or**

```
FaxTransportMode = 1 (Relay)
V34ModemTransportType = 2 (Modem bypass)
```

In this configuration, T.30 faxes use Relay mode (T.38) while V.34 fax uses Bypass mode.

In order to use V.34 fax in Relay mode (T.38), you must configure:

```
FaxTransportMode = 1 (Relay)
V34ModemTransportType = 0 (Transparent)
V32ModemTransportType = 0
V23ModemTransportType = 0
V22ModemTransportType = 0
```

This configuration forces the V.34 fax machine to work in T.30 mode.

## Appendix E Mediant 2000 Clock Settings

The gateway can either generate its own timing signals, using an internal clock, or recover them from one of the E1/T1 trunks.

**a. To use the internal gateway clock source configure the following parameters:**

- TDMBusClockSource = 1
- ClockMaster = 1 (for all gateway trunks)

**b. To use the recovered clock option configure the following parameters:**

- TDMBusClockSource = 4
- ClockMaster\_x = 0 (for all "slave" gateway trunks connected to PBX#1)
- ClockMaster\_x = 1 (for all "master" gateway trunks connected to PBX#2)

Assuming that the gateway recovers its internal clock from one of the "slave" trunks connected to PBX#1, and provides clock to PBX#2 on its "master" trunks.

In addition it is necessary to define from which of the "slave" trunks the gateway recovers its clock:

- TDMBusPSTNAutoClockEnable = 1 (the gateway automatically selects one of the connected "slave" trunks)

Or

- TDMBusLocalReference = # (Trunk index: 0 to 7, default = 0)



**Note 1:** To configure the TDM Bus Clock Source parameters, refer to Section 5.9.4 on page 68.

**Note 2:** When the Mediant 2000 is used in a 'non-span' configuration, the internal gateway clock must be used (as explained above).

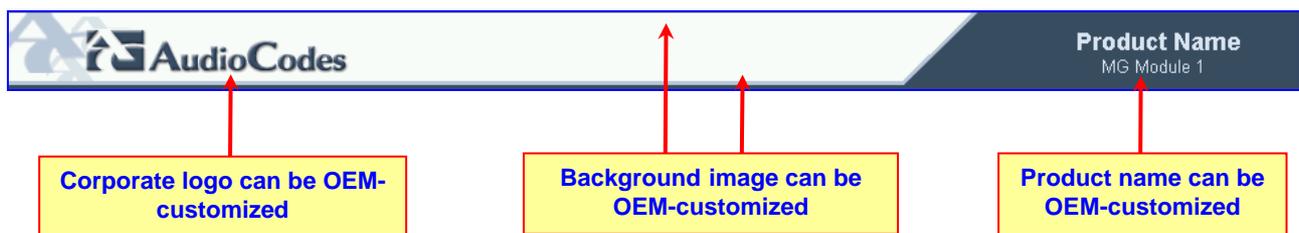
## Reader's Notes

## Appendix F Customizing the Mediant 2000 Web Interface

Customers incorporating the Mediant 2000 into their portfolios can customize the device's Web Interface to suit their specific corporate logo and product naming conventions.

Customers can customize the Web Interface's title bar (AudioCodes' title bar is shown in [Figure F-1](#); a customized title bar is shown in [Figure F-3](#)).

**Figure F-1: User-Customizable Web Interface Title Bar**



**Figure F-2: Customized Web Interface Title Bar**



➤ **To customize the title bar via the Web Interface, take these 3 steps:**

1. Replace the main corporate logo (refer to [Section F.1](#) below).
2. Replace the title bar's background image file (refer to [Section F.2](#) on page 209).
3. Customize the product's name (refer to [Section F.3](#) on page 210).

### F.1 Replacing the Main Corporate Logo

The main corporate logo can be replaced either with a different logo image file (refer to [Section F.1.1](#) below) or with a text string (refer to [Section F.1.2](#) on page 209). Note that when the main corporation logo is replaced, AudioCodes' logo on the left bar (refer to [Figure 5-2](#)) and in the Software Upgrade Wizard (refer to [Section 5.11.1](#) on page 78) disappear.

Also note that the browser's title bar is automatically updated with the string assigned to the WebLogoText parameter when AudioCodes' default logo is not used.

#### F.1.1 Replacing the Main Corporate Logo with an Image File



**Note:** Use a gif, jpg or jpeg file for the logo image. It is important that the image file has a fixed height of 59 pixels (the width can be configured). The combined size of the image files (logo and background) is limited to 64 kbytes.

➤ **To replace the default logo with your own corporate image via the Web Interface, take these 7 steps:**

1. Access the Mediant 2000 Embedded Web Server (refer to [Section 5.6](#) on page 41).

2. In the URL field, append the suffix 'AdminPage' (note that it's case-sensitive) to the IP address, e.g., <http://10.1.229.17/AdminPage>.
3. Click **Image Load to Device**; the Image Download screen is displayed (shown in [Figure F-3](#)).

**Figure F-3: Image Download Screen**

Send "Logo Image" file from your computer to the device

---

Send "Background Image" file from your computer to the device

---

Logo width

---

This button restores the default images

**Important!**  
Use the 'Save Configuration' Link in order to save loaded images to flash memory

4. Click the **Browse** button in the **Send Logo Image File from your computer to the Device** box. Navigate to the folder that contains the logo image file you want to load.
5. Click the **Send File** button; the file is sent to the device. When loading is complete, the screen is automatically refreshed and the new logo image is displayed.
6. Note the appearance of the logo. If you want to modify the width of the logo (the default width is 339 pixels), in the **Logo Width** field, enter the new width (in pixels) and press the **Set Logo Width** button.
7. To save the image to flash memory so it is available after a power fail, refer to [Section 5.12](#) on page 84.

The new logo appears on all Web Interface screens.



**Tip:** If you encounter any problem during the loading of the files, or you want to restore the default images, click the **Restore Default Images** button.

➤ **To replace the default logo with your own corporate image via the *ini* file, take these 2 steps:**

1. Place your corporate logo image file in the same folder as where the device's *ini* file is located (i.e., the same location defined in the BootP/TFTP configuration utility). For detailed information on the BootP/TFTP, refer to [Appendix B](#) on page 189.
2. Add/modify the two *ini* file parameters in [Table F-1](#) according to the procedure described in [Section 6.2](#) on page 87.

Note that loading the device's *ini* file via the 'Configuration File' screen in the Web Interface doesn't load the corporate logo image files as well.

Table F-1: Customizable Logo *ini* File Parameters

Parameter	Description
LogoFileName	The name of the image file containing your corporate logo. Use a gif, jpg or jpeg image file. The default is AudioCodes' logo file. <b>Note:</b> The length of the name of the image file is limited to 47 characters.
LogoWidth	Width (in pixels) of the logo image. <b>Note:</b> The optimal setting depends on the resolution settings. The default value is 339, which is the width of AudioCodes' displayed logo.

## F.1.2 Replacing the Main Corporate Logo with a Text String

The main corporate logo can be replaced with a text string.

- To replace AudioCodes' default logo with a text string *via the Web Interface*, modify the two *ini* file parameters in Table F-2 according to the procedure described in Section F.4 on page 211.
- To replace AudioCodes' default logo with a text string *via the ini file*, add/modify the two *ini* file parameters in Table F-2 according to the procedure described in Section 6.2 on page 87.

Table F-2: Web Appearance Customizable *ini* File Parameters

Parameter	Description
UseWebLogo	0 = Logo image is used (default). 1 = Text string is used instead of a logo image.
WebLogoText	Text string that replaces the logo image. The string can be up to 15 characters.

## F.2 Replacing the Background Image File

The background image file is duplicated across the width of the screen. The number of times the image is duplicated depends on the width of the background image and screen resolution. When choosing your background image, keep this in mind.



**Note:** Use a gif, jpg or jpeg file for the background image. It is important that the image file has a fixed height of 59 pixels. The combined size of the image files (logo and background) is limited to 64 kbytes.

### ➤ To replace the background image via the Web, take these 6 steps:

1. Access the Mediant 2000 Embedded Web Server (refer to Section 5.6 on page 41).
2. In the URL field, append the suffix 'AdminPage' (note that it's case-sensitive) to the IP address, e.g., `http://10.1.229.17/AdminPage`.
3. Click the **Image Load to Device**, the Image Download screen is displayed (shown in Figure F-3).
4. Click the **Browse** button in the **Send Background Image File from your computer to gateway** box. Navigate to the folder that contains the background image file you want to load.
5. Click the **Send File** button; the file is sent to the device. When loading is complete, the screen is automatically refreshed and the new background image is displayed.

6. To save the image to flash memory so it is available after a power fail, refer to Section 5.12 on page 84.

The new background appears on all Web Interface screens.



**Tip 1:** If you encounter any problem during the loading of the files, or you want to restore the default images, click the **Restore Default Images** button.

**Tip 2:** When replacing both the background image and the logo image, first load the logo image followed by the background image.

➤ **To replace the background image via the *ini* file, take these 2 steps:**

1. Place your background image file in the same folder as where the device's *ini* file is located (i.e., the same location defined in the BootP/TFTP configuration utility). For detailed information on the BootP/TFTP, refer to Appendix B on page 189.
2. Add/modify the *ini* file parameters in Table F-3 according to the procedure described in Section 6.2 on page 87.

Note that loading the device's *ini* file via the 'Configuration File' screen in the Web Interface doesn't load the logo image file as well.

**Table F-3: Customizable Logo *ini* File Parameters**

Parameter	Description
BkgImageFileName	The name (and path) of the file containing the new background. Use a gif, jpg or jpeg image file. The default is AudioCodes background file. <b>Note:</b> The length of the name of the image file is limited to 47 characters.

### F.3 Customizing the Product Name

The Product Name text string can be modified according to OEMs specific requirements.

- To replace AudioCodes' default product name with a text string *via the Web Interface*, modify the two *ini* file parameters in Table F-4 according to the procedure described in Section F.4 on page 211.
- To replace AudioCodes' default product name with a text string *via the ini file*, add/modify the two *ini* file parameters in Table F-4 according to the procedure described in Section 6.2 on page 87.

**Table F-4: Web Appearance Customizable *ini* File Parameters**

Parameter	Description
UseProductName	0 = Don't change the product name (default). 1 = Enable product name change.
UserProductName	Text string that replaces the product name. The default is "Mediant 2000". The string can be up to 29 characters.

## F.4 Modifying *ini* File Parameters via the Web AdminPage

- To modify *ini* file parameters via the AdminPage, take these 6 steps:
1. Access the Mediant 2000 Embedded Web Server (refer to Section 5.6 on page 41).
  2. In the URL field, append the suffix 'AdminPage' (note that it's case-sensitive) to the IP address, e.g., `http://10.1.229.17/AdminPage`.
  3. Click the **INI Parameters** option, the INI Parameters screen is displayed (shown in Figure F-4).

Figure F-4: INI Parameters Screen

Parameter name:  Enter value:

Parameter name:  Enter value:

OUTPUT WINDOW

4. In the **Parameter Name** dropdown list, select the required *ini* file parameter.
5. In the **Enter Value** field to the right, enter the parameter's new value.
6. Click the **Apply new value** button to the right; the INI Parameters screen is refreshed, the parameter name with the new value appears in the fields at the top of the screen and the **Output Window** displays a log displaying information on the operation.



**Note:** You cannot load the image files (e.g., logo/background image files) to the device by choosing a file name parameter in this screen.

## Reader's Notes

## Appendix G Accessory Programs and Tools

The accessory applications and tools shipped with the device provide you with friendly interfaces that enhance device usability and smooth your transition to the new VoIP infrastructure. The following applications are available:

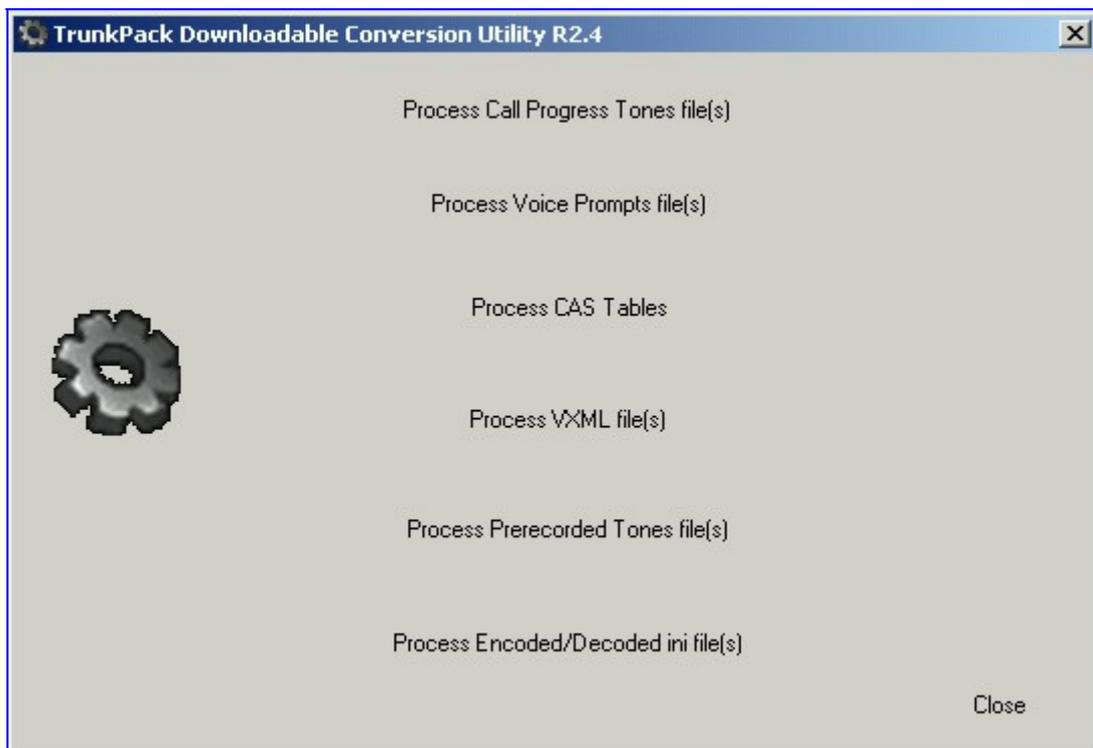
- TrunkPack Downloadable Conversion Utility (refer to Section G.1 below).
- PSTN Trace Utility (refer to Section G.1.4 on page 218).

### G.1 TrunkPack Downloadable Conversion Utility

Use the TrunkPack Downloadable Conversion Utility to:

- Create a loadable Call Progress Tones file (refer to Section G.1.1 on page 214).
- Create a loadable Voice Prompts file from pre-recorded voice messages (refer to Section G.1.2 on page 215).
- Encode / decode an *ini* file (refer to Section G.1.3 on page 217).
- Create a loadable Prerecorded Tones file (refer to Section G.1.3 on page 217).

**Figure G-1: TrunkPack Downloadable Conversion Utility Opening Screen**



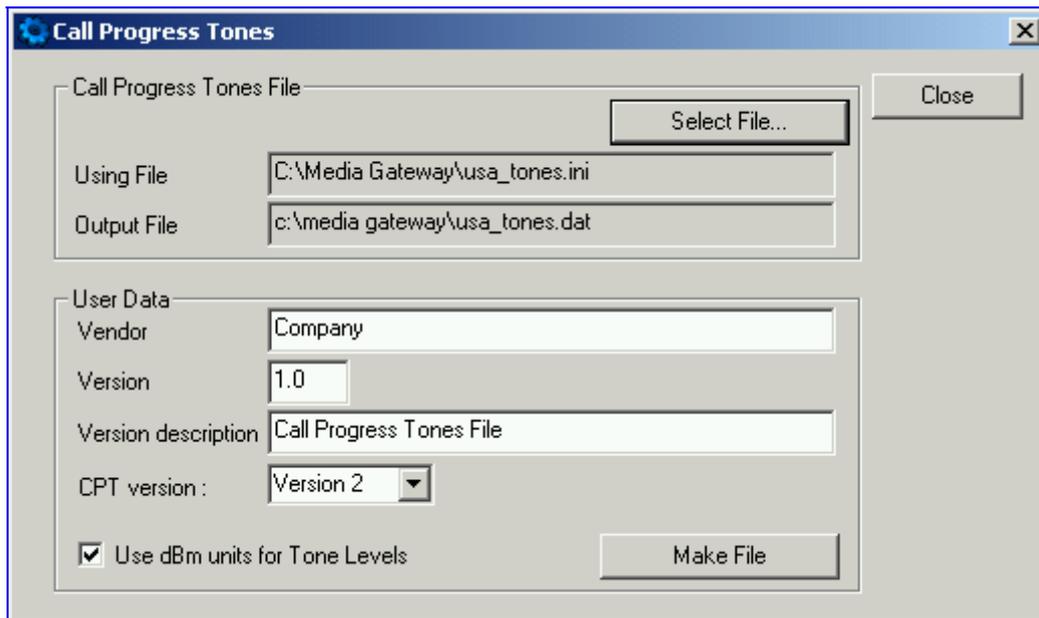
### G.1.1 Converting a CPT *ini* File to a Binary *dat* File

For detailed information on creating a CPT *ini* file, refer to Section 7.1 on page 135.

➤ **To convert a CPT *ini* file to a binary *dat* file, take these 10 steps:**

1. Execute the TrunkPack Downloadable Conversion Utility, DConvert240.exe (supplied with the software package); the utility's main screen opens (shown in Figure G-1).
2. Click the **Process Call Progress Tones File(s)** button; the Call Progress Tones screen, shown in Figure G-2, opens.

**Figure G-2: Call Progress Tones Conversion Screen**



3. Click the **Select File...** button that is in the 'Call Progress Tone File' box.
4. Navigate to the folder that contains the CPT *ini* file you want to convert.
5. Click the *ini* file and click the **Open** button; the name and path of both the *ini* file and the (output) *dat* file appears in the fields below the Select File button.
6. Enter the Vendor Name, Version Number and Version Description in the corresponding required fields under the 'User Data' section.
7. Set 'CPT Version' to 'Version 1' only if you use this utility with a version released before version 4.4 of the device software (this field is used to maintain backward compatibility).
8. Check the 'Use dBm units for Tone Levels' check box. Note that the levels of the Call Progress Tones (in the CPT file) must be in -dBm units.
9. Click the **Make File** button; you're prompted that the operation (conversion) was successful.
10. Close the application.

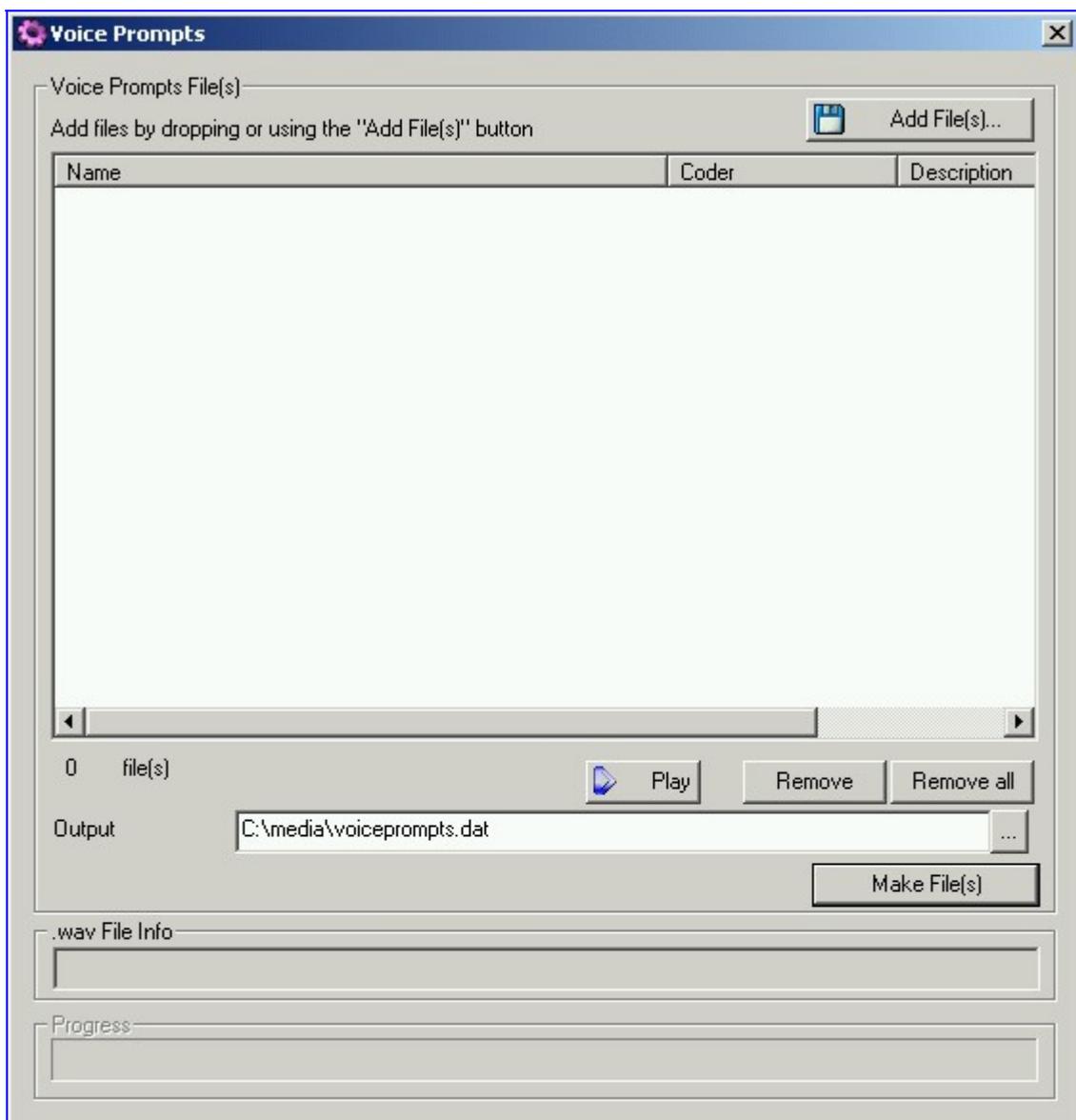
## G.1.2 Creating a Loadable Voice Prompts File

For detailed information on the Voice Prompts file, refer to Section 7.2 on page 137.

➤ **To create a loadable Voice Prompts *.dat* file from your voice recording files, take these 7 steps:**

1. Execute the TrunkPack Downloadable Conversion Utility, DConvert240.exe (supplied with the software package); the utility's main screen opens (shown in Figure G-1).
2. Click the **Process Voice Prompts File(s)** button; the Voice Prompts screen, shown in Figure G-3, opens.

**Figure G-3: Voice Prompts Screen**



3. To add the pre-recorded voice files to the 'Voice Prompts' screen follow one of these procedures:
  - Select the files and drag them to the 'Voice Prompts' screen.
  - Click the **Add File(s)** button; the 'Select Files' screen opens. Select the required Voice Prompt files and press the **Add>>** button. Close the 'Select Files' screen.

4. Arrange the files according to your requirements by dragging and dropping them from one location in the list to another. Note that the sequence of the files determines their assigned Voice Prompt ID.



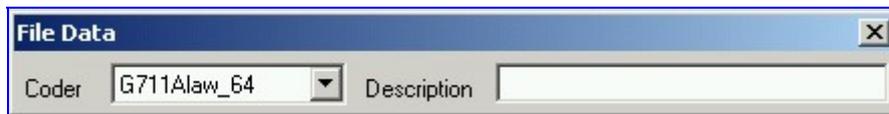
**Tip 1:** Use the **Play** button to play *wav* files through your PC speakers.

**Tip 2:** Use the **Remove** and **Remove all** buttons to delete files from the list.

5. For each of the raw files, select a coder that corresponds with the coder it was *originally* recorded in by completing the following steps:
  - Double-click or right-click the required file(s); the 'File Data' window (shown in [Figure G-4](#)) appears.
  - From the 'Coder' drop-down list, select the required coder type.
  - In the 'Description' field, enter additional identifying information.
  - Close the 'File Data' window.

Note that for *wav* files, a coder is automatically selected from the *wav* file's header.

**Figure G-4: File Data Window**



6. In the 'Output' field, specify the output directory in which the Voice Prompts file is generated followed by the name of the Voice Prompts file (the default name is *voiceprompts.dat*).
7. Press the **Make File(s)** button; the Voice Prompts loadable file is produced.

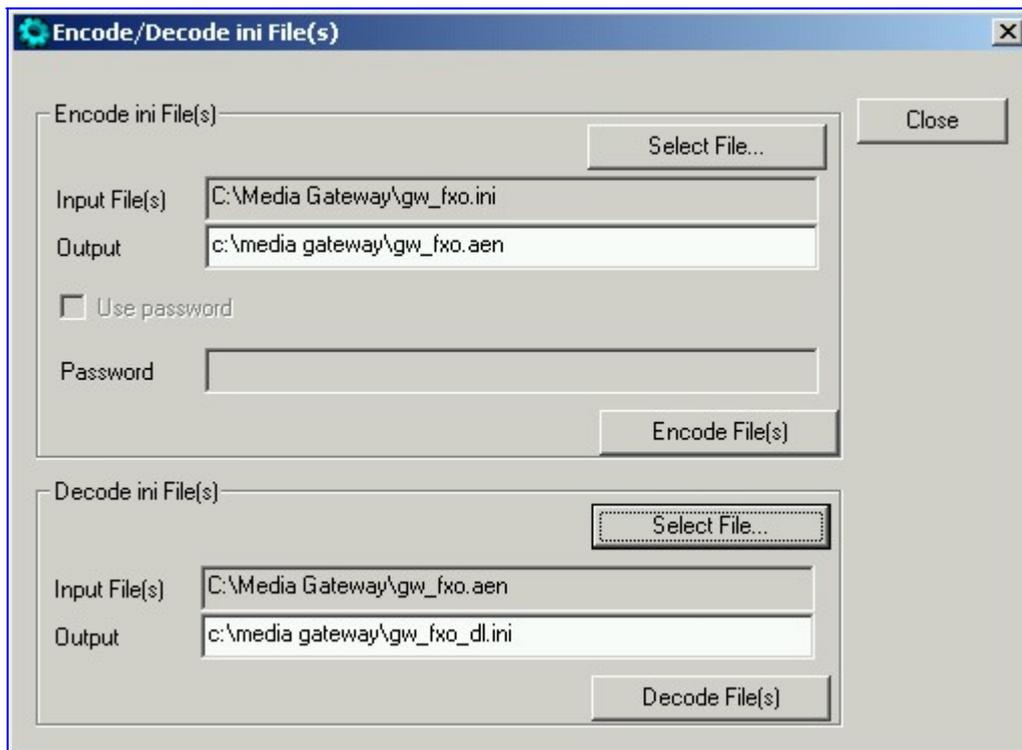
### G.1.3 Encoding / Decoding an *ini* File

For detailed information on secured *ini* file, refer to Section 6.1 on page 87.

➤ **To encode an *ini* file, take these 6 steps:**

1. Execute the TrunkPack Downloadable Conversion Utility, DConvert240.exe (supplied with the software package); the utility's main screen opens (shown in Figure G-1).
2. Click the **Process Encoded/Decoded *ini* file(s)** button; the 'Encode/Decode *ini* File(s)' screen, shown in Figure G-5, opens.

Figure G-5: Encode/Decode *ini* File(s) Screen



3. Click the **Select File...** button under the 'Encode *ini* File(s)' section.
4. Navigate to the folder that contains the *ini* file you want to encode.
5. Click the *ini* file and click the **Open** button; the name and path of both the *ini* file and the output encoded file appear in the fields under the **Select File** button. Note that the name and extension of the output file can be modified.
6. Click the **Encode File(s)** button; an encoded *ini* file with the name and extension you specified is created.

➤ **To decode an encoded *ini* file, take these 4 steps:**

1. Click the **Select File...** button under the 'Decode *ini* File(s)' section.
2. Navigate to the folder that contains the file you want to decode.
3. Click the file and click the **Open** button. the name and path of both the encode *ini* file and the output decoded file appear in the fields under the **Select File** button. Note that the name of the output file can be modified.
4. Click the **Decode File(s)** button; a decoded *ini* file with the name you specified is created.

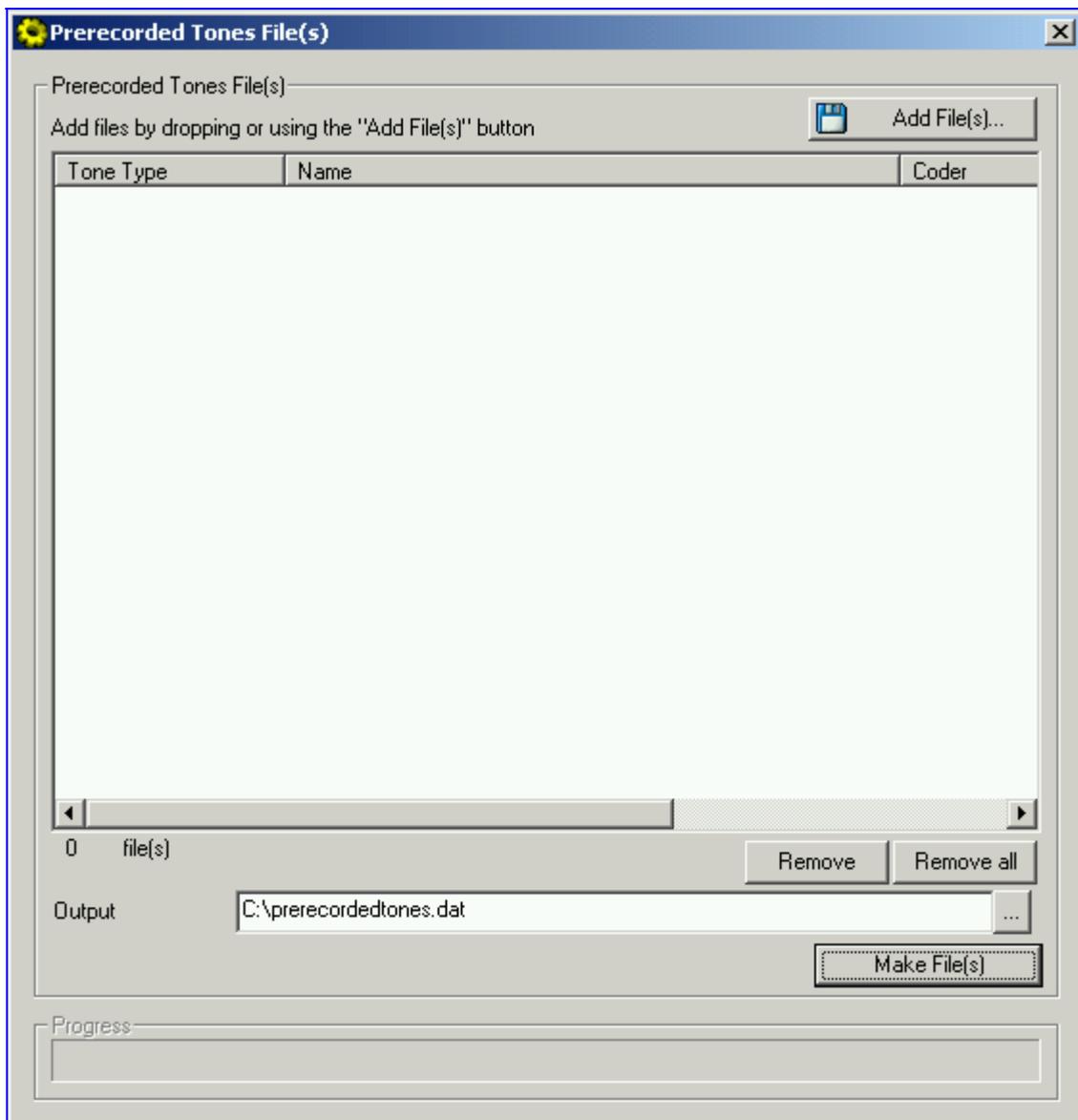
Note that the decoding process verifies the input file for validity. Any change made to the encoded file causes an error and the decoding process is aborted.

## G.1.4 Creating a Loadable Prerecorded Tones File

For detailed information on the PRT file, refer to Section 7.2 on page 137.

- **To create a loadable PRT *dat* file from your raw data files, take these 7 steps:**
  1. Prepare the prerecorded tones (raw data PCM or L8) files you want to combine into a single *dat* file using standard recording utilities.
  2. Execute the TrunkPack Downloadable Conversion utility, DConvert240.exe (supplied with the software package); the utility's main screen opens (shown in Figure G-1).
  3. Click the **Process Prerecorded Tones File(s)** button; the Prerecorded Tones File(s) screen, shown in Figure G-3, opens.

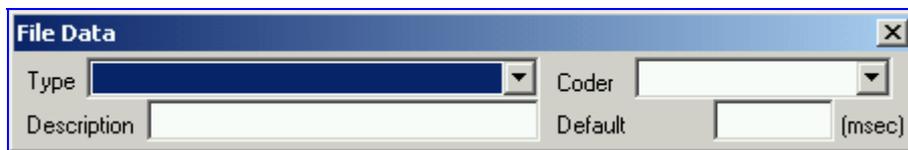
**Figure G-6: Prerecorded Tones Screen**



4. To add the prerecorded tone files (you created in Step 1) to the 'Prerecorded Tones' screen follow one of these procedures:
  - Select the files and drag them to the 'Prerecorded Tones' screen.

- Click the **Add File(s)** button; the 'Select Files' screen opens. Select the required Prerecorded Tone files and press the **Add>>** button. Close the 'Select Files' screen.
5. For each raw data file, define a Tone Type, a Coder and a Default Duration by completing the following steps:
- Double-click or right-click the required file; the 'File Data' window (shown in [Figure G-4](#)) appears.
  - From the 'Type' drop-down list, select the tone type this raw data file is associated with.
  - From the 'Coder' drop-down list, select the coder that corresponds to the coder this raw data file was *originally* recorded with.
  - In the 'Description' field, enter additional identifying information (optional).
  - In the 'Default' field, enter the default duration this raw data file is repeatedly played.
  - Close the 'File Data' window (press the **Esc** key to cancel your changes); you are returned to the Prerecorded Tones File(s) screen.

**Figure G-7: File Data Window**



6. In the 'Output' field, specify the output directory in which the PRT file is generated followed by the name of the PRT file (the default name is *prerecordedtones.dat*). Alternatively, use the Browse button to select a different output file. Navigate to the desired file and select it; the selected file name and its path appear in the 'Output' field.
7. Click the **Make File(s)** button; the Progress bar at the bottom of the window is activated. The *dat* file is generated and placed in the directory specified in the 'Output' field. A message box informing you that the operation was successful indicates that the process is completed.

## G.2 PSTN Trace Utility

These utilities are designed to convert PSTN trace binary files to textual form. The binary PSTN trace files are generated when the User sets the PSTN interface to trace mode.

### G.2.1 Operation

#### Generating textual trace/audit file for CAS protocols -

To generate a readable text file out of the binary trace file when using CAS protocols, rename the PSTN trace binary file to `CASTrace0.dat` and copy it to the same directory in which the translation utility `CAS_Trace.exe` is located. Then, run `CAS_Trace.exe` (no arguments are required). As a result, the textual file `CASTrace0.txt` is created.

#### Generating textual trace/audit file for ISDN PRI protocols -

To generate a readable text file out of the binary trace file when using ISDN protocols, copy the PSTN trace binary file to the same directory in which the translation utility `Convert_Trace.bat` is located. The following files should reside in the same directory: `Dumpview.exe`, `Dumpview.cfg` and `ReadMe.txt`. Please read carefully the `ReadMe.txt` in order to understand the usage of the translation utility. Next, run the `Convert_Trace.bat`. As a result, the textual file is created.

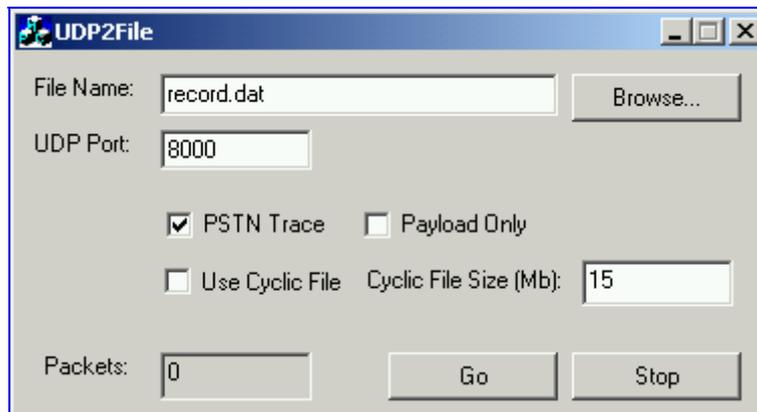
To start and collect the PSTN trace via the Web, please use the following instructions. (Refer to [Figure H-8](#) for a view of the Trunk Traces). Also, please note if the PSTN trace was of a PRI or CAS collection based on the framer involved in the trace. This information is needed to properly parse the captured data.

1. Run the UDP2File utility.
2. Determine the trace file name.
3. Determine the UDP port.
4. Mark the PSTN Trace check box.
5. Push the Run button=> the UDP2File utility starts to collect the trace messages.
6. Activate the Web page by entering <Mediant 2000 IP address>/TrunkTraces, such as: `http://10.8.8.101/TrunkTraces`. The user and password is the same for the unit.
7. In the Web page set the trace level of each trunk.
8. Enable the trace via the Web.
9. Determine the UDP port (the same as in step 3).
10. Push the Submit button => the board starts to send the trace messages. In the UDP2File utility (Refer to [Figure H-9](#)) you should see the number in the packets counter increasing.

**Figure H-8: Trunk Traces**

Trunk Traces	
Trace Level Trunk 1	acFULL_TRACE
Trace Level Trunk 2	acLAYER3_ISDN_TRACE
Trace Level Trunk 3	acFULL_TRACE
Trace Level Trunk 4	acLAYER3_ISDN_TRACE_No_Duplicatic
Trace Level Trunk 5	acNO_TRACE
Trace Level Trunk 6	acNO_TRACE
Trace Level Trunk 7	acNO_TRACE
Trace Level Trunk 8	acNO_TRACE
Enable Pstn Trace from Web	On
Port	8000

**Figure H-9: UDP2File Utility**



## Reader's Notes

## Appendix H Software Upgrade Key

### H.1 About the Software Upgrade Key

Mediant 2000 gateways are supplied with a Software Upgrade Key already pre-configured for each of its TrunkPack Modules (TPM).

Users can later upgrade their Mediant 2000 features, capabilities and quantity of available resources by specifying what upgrades they require, and purchasing a new key to match their specification.

The Software Upgrade Key is sent as a string in a text file, to be loaded into the Mediant 2000. Stored in the device's non-volatile flash memory, the string defines the features and capabilities allowed by the specific key purchased by the user. The device allows users to utilize *only these* features and capabilities. A new key overwrites a previously installed key.



**Note:** The Software Upgrade Key is an encrypted key. Each TPM utilizes a unique key. The Software Upgrade Key is provided by AudioCodes only.

### H.2 Backing up the Current Software Upgrade Key

Back up your current Software Upgrade Key before loading a new key to the device. You can always reload this backed-up key to restore your device capabilities to what they originally were if the 'new' key doesn't comply with your requirements.

➤ **To backup the current Software Upgrade Key, take these 5 steps:**

1. Access the devices Embedded Web Server (refer to Section 5.5 on page 40).
2. Click the **Software Update** button.
3. Click the **Software Upgrade Key** tab; the Software Upgrade Key screen is displayed (shown in Figure H-1).
4. Copy the string from the **Current Key** field and paste it in a new file.
5. Save the text file with a name of your choosing.

### H.3 Loading the Software Upgrade Key

After receiving the Software Upgrade Key file (do not modify its contents in any way), ensure that its first line is [LicenseKeys] and that it contains one or more lines in the following format:

S/N<Serial Number of TPM> = <long Software Upgrade Key>

For example: S/N370604 = jCx6r5tovCIKaBBbhPtT53Yj...

One S/N must match the S/N of your device. The device's S/N can be viewed in the 'Device Information' screen (refer to Section 5.10.4 on page 77).

You can load a Software Upgrade Key using:

- The Embedded Web Server (refer to Section H.3.1 below).
- The BootP/TFTP configuration utility (refer to Section H.3.2 on page 225).
- AudioCodes' EMS (refer to Section 11.8 on page 182 and to AudioCodes' EMS User's Manual or EMS Product Description).

### H.3.1 Loading the Software Upgrade Key Using the Embedded Web Server

➤ **To load a Software Upgrade Key using the Web Server, take these 5 steps:**

1. Access the devices Embedded Web Server (refer to Section 5.5 on page 40).
2. Click the **Software Update** button.
3. Click the **Software Upgrade Key** tab; the Software Upgrade Key screen is displayed (shown in Figure H-1).
4. When loading a single key S/N line to a device:  
Open the Software Upgrade Key file (it should open in Notepad), select and copy the key string of the device's S/N and paste it into the Web field **New Key**. If the string is sent in the body of an email, copy and paste it from there. Press the **Add Key** button.

When loading a Software Upgrade Key text file containing multiple S/N lines to a device (refer to Figure H-2):

Click the **Browse** button in the **Send "Upgrade Key" file from your computer to the device** field, and navigate to the Software Upgrade Key text file.  
Click the **Send File** button.

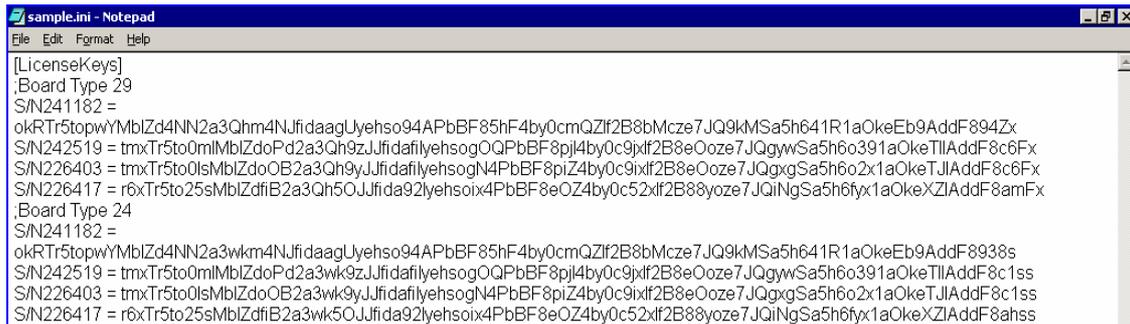
The new key is loaded to the device, validated and if valid is burned to memory. The new key is displayed in the **Current Key** field.

Validate the new key by scrolling through the 'Key features:' panel and verifying the presence / absence of the appropriate features.

5. After verifying that the Software Upgrade Key was successfully loaded, reset the device; the new capabilities and resources are active.

**Figure H-1: Software Upgrade Key Screen**



**Figure H-2: Example of a Software Upgrade Key File Containing Multiple S/N Lines**


```

sample.ini - Notepad
File Edit Format Help
[LicenseKeys]
;Board Type 29
S/N241182 =
okRTr5topwYMblZd4NN2a3Qhm4Njfi daagUyehso94APbBF85hF4by0cmQZlf2B8bMcze7JQ9kMSa5h641R1aOkeEb9AddF894Zx
S/N242519 = tmxTr5to0mlMblZdoPd2a3Qh9zJfidafileyehsogOQPbBF8pi4by0c9jxf2B8eOoze7JQgywSa5h6o391aOkeTIIAddF8c6Ffx
S/N226403 = tmxTr5to0lsMblZdoB2a3Qh9yJfidafileyehsogN4PbBF8piZ4by0c9jxf2B8eOoze7JQgwgSa5h6o2x1aOkeTIIAddF8c6Ffx
S/N226417 = r6xTr5to25sMblZdfiB2a3Qh5OJfida92lyehsoix4PbBF8eOZ4by0c52xf2B88yoze7JQiNgSa5h6fyx1aOkeXZIIAddF8amFfx
;Board Type 24
S/N241182 =
okRTr5topwYMblZd4NN2a3wkm4Njfi daagUyehso94APbBF85hF4by0cmQZlf2B8bMcze7JQ9kMSa5h641R1aOkeEb9AddF8938s
S/N242519 = tmxTr5to0mlMblZdoPd2a3wk9zJfidafileyehsogOQPbBF8pi4by0c9jxf2B8eOoze7JQgywSa5h6o391aOkeTIIAddF8c1ss
S/N226403 = tmxTr5to0lsMblZdoB2a3wk9yJfidafileyehsogN4PbBF8piZ4by0c9jxf2B8eOoze7JQgwgSa5h6o2x1aOkeTIIAddF8c1ss
S/N226417 = r6xTr5to25sMblZdfiB2a3wk5OJfida92lyehsoix4PbBF8eOZ4by0c52xf2B88yoze7JQiNgSa5h6fyx1aOkeXZIIAddF8ahss

```

### H.3.2 Loading the Software Upgrade Key Using BootP/TFTP

- **To load the Software Upgrade Key file using BootP/TFTP, take these 5 steps:**

1. Place the file in the same location you've saved the *device's cmp* file.
2. Start the BootP/TFTP configuration utility and from the **Services** menu in the main screen, choose option **Clients**; the Client Configuration screen is displayed (refer to Figure [Figure B-4](#) on page 195).
3. From the drop-down list in the **INI File** field, select the Software Upgrade Key file instead of the device's *ini* file. Note that the device's *cmp* file must be specified in the **Boot File** field.
4. Configure the initial BootP/TFTP parameters required, and click **OK** (refer to [Appendix B](#) on page 189).
5. Reset the device; the device's *cmp* and Software Upgrade Key files are loaded to the device.

## H.4 Verifying that the Key was Successfully Loaded

After installing the key, you can determine in the Embedded Web Server's read-only 'Key features:' panel (**Software Update** menu > **Software Upgrade Key**) (refer to [Figure H-1](#)) that the features and capabilities activated by the installed string match those that were ordered.

You can also verify that the key was successfully loaded to the device by accessing the Syslog server. For detailed information on the Syslog server, refer to [Section 9.2](#) on page 165. When a key is successfully loaded, the following message is issued in the Syslog server:

"S/N\_\_\_ Key Was Updated. The Board Needs to be Reloaded with *ini* file\n"

## H.5 Troubleshooting an Unsuccessful Loading of a Key

If the Syslog server indicates that a Software Upgrade Key file was unsuccessfully loaded (the SN\_ line is blank), take the following preliminary actions to troubleshoot the issue:

- Open the Software Upgrade Key file and check that the S/N line of the specific device whose key you want to update is listed in it. If it isn't, contact AudioCodes.
- Verify that you've loaded the correct file and that you haven't loaded the device's *ini* file or the CPT *ini* file by mistake. Open the file and ensure that the first line is [LicenseKeys].
- Verify that you didn't alter in any way the contents of the file.

## H.6 Abort Procedure

Reload the key you backed-up in [Section Backing up the Current Software Upgrade Key](#) on page 223 to restore your device capabilities to what they originally. To load the backed-up key use the procedure described in [Section Loading the Software Upgrade Key](#) on page 223.

## Reader's Notes

## Appendix I Release Reason Mapping

Table I-1 below describes the mapping of ISDN release reason to SIP response. Table I-2 on page 229 describes the mapping of SIP response to ISDN release reason.

**Table I-1: Mapping of ISDN Release Reason to SIP Response (continues on pages 227 to 228)**

ISDN Release Reason	Description	SIP Response	Description
1	Unallocated number	404	Not found
2	No route to network	404	Not found
3	No route to destination	404	Not found
6	Channel unacceptable	406*	Not acceptable
7	Call awarded and being delivered in an established channel	500	Server internal error
16	Normal call clearing	-*	BYE
17	User busy	486	Busy here
18	No user responding	408	Request timeout
19	No answer from the user	480	Temporarily unavailable
20	Subscriber absent	480	Temporarily unavailable
21	Call rejected	403	Forbidden
22	Number changed w/o diagnostic	410	Gone
22	Number changed with diagnostic	410	Gone
23	Redirection to new destination	480	Temporarily unavailable
26	Non-selected user clearing	404	Not found
27	Destination out of order	502	Bad gateway
28	Address incomplete	484	Address incomplete
29	Facility rejected	501	Not implemented
30	Response to status enquiry	501*	Not implemented
31	Normal unspecified	480	Temporarily unavailable
34	No circuit available	503	Service unavailable
38	Network out of order	503	Service unavailable
41	Temporary failure	503	Service unavailable
42	Switching equipment congestion	503	Service unavailable
43	Access information discarded	502*	Bad gateway
44	Requested channel not available	503*	Service unavailable
47	Resource unavailable	503	Service unavailable
49	QoS unavailable	503*	Service unavailable
50	Facility not subscribed	503*	Service unavailable
55	Incoming calls barred within CUG	403	Forbidden
57	Bearer capability not authorized	403	Forbidden
58	Bearer capability not presently available	503	Service unavailable
63	Service/option not available	503*	Service unavailable
65	Bearer capability not implemented	501	Not implemented

\* Messages and responses were created since the 'ISUP to SIP Mapping' draft doesn't specify their cause code mapping.

ISDN Release Reason	Description	SIP Response	Description
66	Channel type not implemented	480*	Temporarily unavailable
69	Requested facility not implemented	503*	Service unavailable
70	Only restricted digital information bearer capability is available	503*	Service unavailable
79	Service or option not implemented	501	Not implemented
81	Invalid call reference value	502*	Bad gateway
82	Identified channel does not exist	502*	Bad gateway
83	Suspended call exists, but this call identity does not	503*	Service unavailable
84	Call identity in use	503*	Service unavailable
85	No call suspended	503*	Service unavailable
86	Call having the requested call identity has been cleared	408*	Request timeout
87	User not member of CUG	503	Service unavailable
88	Incompatible destination	503	Service unavailable
91	Invalid transit network selection	502*	Bad gateway
95	Invalid message	503	Service unavailable
96	Mandatory information element is missing	409*	Conflict
97	Message type non-existent or not implemented	480*	Temporarily not available
98	Message not compatible with call state or message type non-existent or not implemented	409*	Conflict
99	Information element non-existent or not implemented	480*	Not found
100	Invalid information elements contents	501*	Not implemented
101	Message not compatible with call state	503*	Service unavailable
102	Recovery of timer expiry	408	Request timeout
111	Protocol error	500	Server internal error
127	Interworking unspecified	500	Server internal error

Table I-2: Mapping of SIP Response to ISDN Release Reason

SIP Response	Description	ISDN Release Reason	Description
400*	Bad request	31	Normal, unspecified
401	Unauthorized	21	Call rejected
402	Payment required	21	Call rejected
403	Forbidden	21	Call rejected
404	Not found	1	Unallocated number
405	Method not allowed	63	Service/option unavailable
406	Not acceptable	79	Service/option not implemented
407	Proxy authentication required	21	Call rejected
408	Request timeout	102	Recovery on timer expiry
409	Conflict	41	Temporary failure
410	Gone	22	Number changed w/o diagnostic
411	Length required	127	Interworking
413	Request entity too long	127	Interworking
414	Request URI too long	127	Interworking
415	Unsupported media type	79	Service/option not implemented
420	Bad extension	127	Interworking
480	Temporarily unavailable	18	No user responding
481*	Call leg/transaction doesn't exist	127	Interworking
482*	Loop detected	127	Interworking
483	Too many hops	25	Exchange – routing error
484	Address incomplete	28	Invalid number format
485	Ambiguous	1	Unallocated number
486	Busy here	17	User busy
488	Not acceptable here	31	Normal, unspecified
500	Server internal error	41	Temporary failure
501	Not implemented	38	Network out of order
502	Bad gateway	38	Network out of order
503	Service unavailable	41	Temporary failure
504	Server timeout	102	Recovery on timer expiry
505*	Version not supported	127	Interworking
600	Busy everywhere	17	User busy
603	Decline	21	Call rejected
604	Does not exist anywhere	1	Unallocated number
606*	Not acceptable	38	Network out of order

\* Messages and responses were created since the 'ISUP to SIP Mapping' draft doesn't specify their cause code mapping.

## Reader's Notes

# Appendix J SS7 Tunneling

The Signaling System 7 (SS7) tunneling feature facilitates peer-to-peer transport of SS7 links between gateways that support AudioCodes' unique MTP2 (Message Transfer Part) Tunneling application (M2TN) for transferring SS7 MTP2 link data over IP. In this scenario, both sides of the link are pure TDM switches and are unaware of the IP tandem that is utilized between them. Using M2TN, the network operator can support SS7 connections over IP, carrying MTP level 3, as well as higher level SS7 layers (e.g., user parts and application protocols, such as TUP (Telephone User Part), Integrated ISUP (Services User Part), SCCP (Signaling Connection Control Part), TCAP (Transaction Capabilities Application Part), etc.).

M2TN uses standard protocols, such as SIGTRAN (RFC 2719 Architectural Framework for Signaling Transport), SCTP (RFC 2960, Stream Control Transmission Protocol), M2UA (RFC 3331, MTP2 User Adaptation Layer), the latter being used for transporting SS7-MTP2 signaling information over IP. M2UA architecture is shown in Figure J-1. M2TN architecture is shown in Figure J-2.

Figure J-1: M2UA Architecture

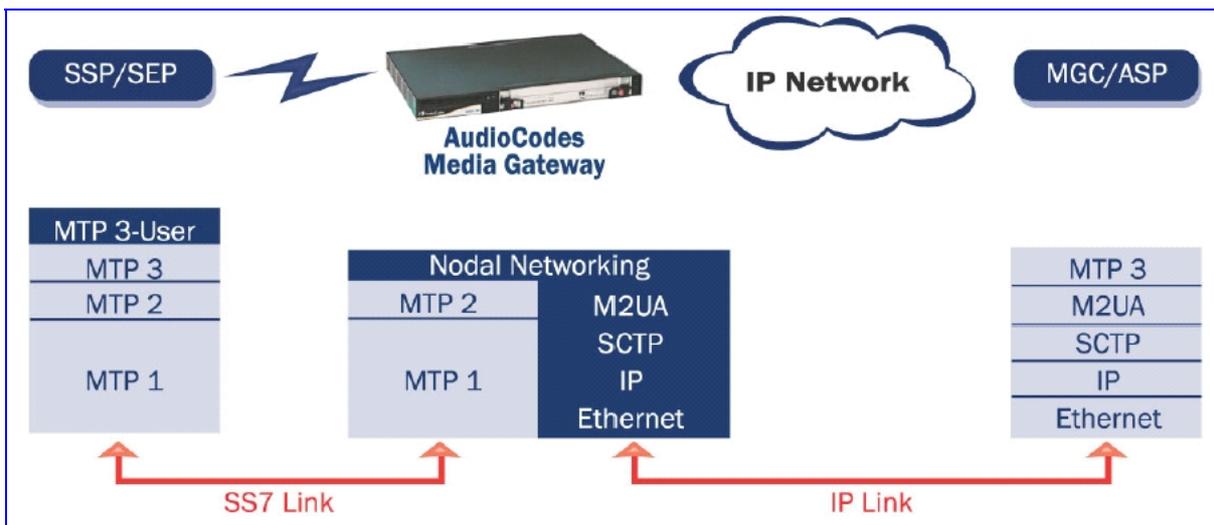
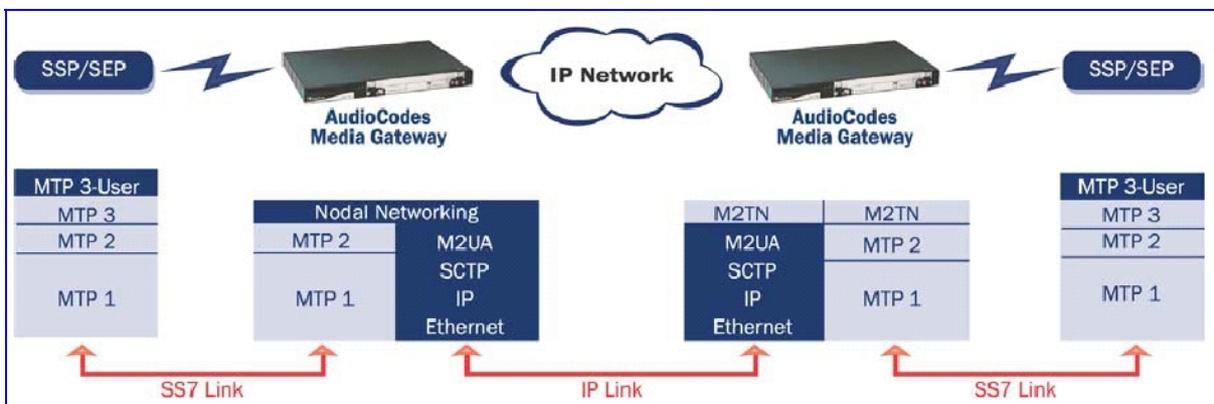


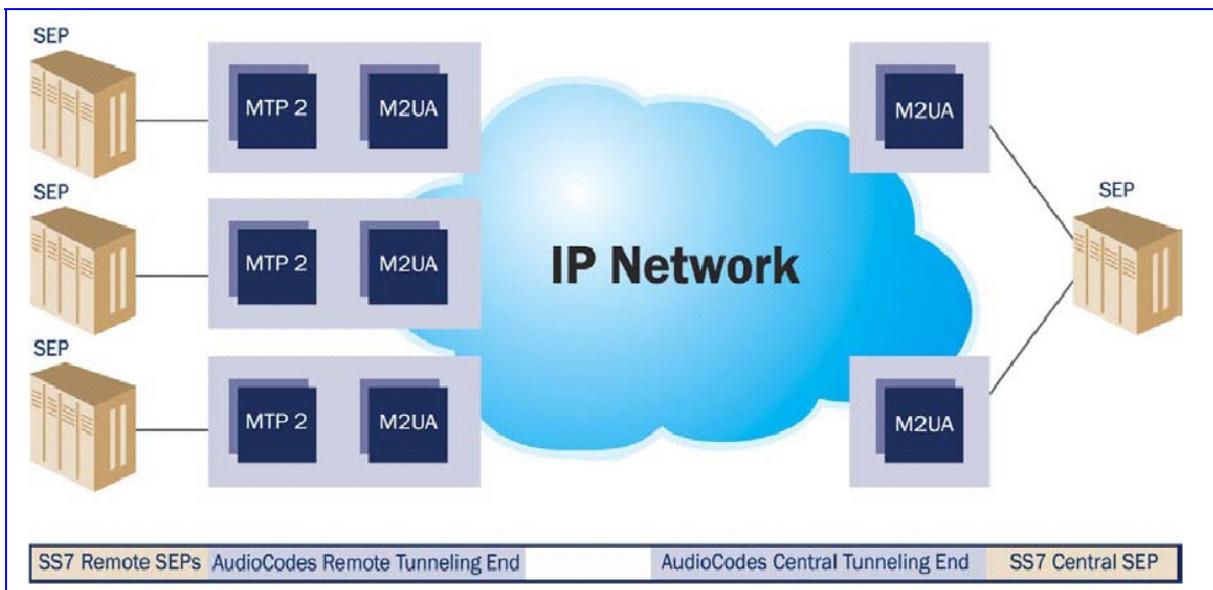
Figure J-2: M2TN Architecture



## J.1 MTP2 Tunneling Technology

The SS7 tunneling technology is based on a pairing of remote and central gateways, as shown in Figure J-3. The remote gateways are configured to backhaul MTP layer 2 signaling over the IP network using standard M2UA protocol (over SCTP protocol). The function of the M2TN entity is to transmit traffic and handle all management events between MTP2 on the TDM side and M2UA's MGC (Media Gateway Controller) entity on the IP side. Only the actual SS7 MSU (Message Signaling Unit) data is sent. Management of the SS7 link is performed using M2UA without transporting the MTP2 LSSU (Link Status Signaling Unit) and FISU (Fill in Signaling Unit) messages over IP. These messages, in addition to MTP2 timing, are terminated and supported, respectively, by the remote and central sides. Therefore, the MTP2 connections are not affected by the fact that they are transported over IP.

Figure J-3: Protocol Architecture for MTP2 Tunneling



## J.2 SS7 Characteristics

- Only standard protocols are used on external interfaces (MTP2 on PSTN side, and M2UA over SCTP on IP side) - the M2TN application resides internally in the Mediant 2000.
- No extra signaling point codes are required; both endpoints are unaware that the SS7 connection is via IP.
- Several links from multiple SS7 nodes can be concentrated into a single board on the 'Central' side (using several SCTP associations per gateway).
- Mediant 2000 gateways can handle SS7 MTP2 tunneling and voice concurrently (does not require additional gateway or other server).
- Voice and signaling can be transferred on the same E1/T1 trunk (F-Links).
- IP traffic can be monitored via standard sniffing tools (e.g. protocol analyzers).

## J.3 SS7 Parameters

The parameters in [Table J-1](#) below configure all MTP attributes simultaneously. To set each MTP attribute individually, add `_xx` (xx equals the element number in the range of 0 to 2) to the end of the `ini` file field name.

**Table J-1: SS7 Parameters (continues on pages 233 to 234)**

<i>ini</i> File Parameter Name	Description
<b>SS7_MTP2_Param_AERM_TIE</b>	Defines the SS7 alignment emergency error rate threshold. The valid range is 0 to 10. The default value is 1.
<b>SS7_MTP2_Param_AERM_TIN</b>	Defines the SS7 alignment normal error rate threshold. The valid range is 0 to 20. The default value is 4.
<b>SS7_MTP2_Param_Error_Correction_Method</b>	Defines the SLI error correction method. 0 = Basic (default) B = Basic P = PCR (Preventive Cyclic Retransmission)
<b>SS7_MTP2_Param_IAC_CP</b>	Defines the number of aborted proving attempts before sending an out-of-service to MTP-3. The valid range is 0 to 10. The default value is 5.
<b>SS7_MTP2_Param_Link_Rate</b>	Defines the SS7 SLI Link Rate. Choose either: 0 = 64 kbps (default) A = 64 kbps D = 56 kbps
<b>SS7_MTP2_Param_LSSU_Length</b>	Defines the SS7 MTP2 LSSU length as 1 or 2 (bytes). The valid range is 1 to 2. The default value is 1.
<b>SS7_MTP2_Param_Octet_Counting</b>	Defines the SS7 MTP2 Octet received while the OCTET is in counting mode (# of Octets received - N Octets - while in Octet counting mode). The valid range is 0 to 256. The default value is 16.
<b>SS7_MTP2_Param_SUERM_SU_D</b>	Defines the SS7 Signal Unit error rate monitor D threshold. The valid range is 0 to 256. The default value is 256.
<b>SS7_MTP2_Param_SUERM_T</b>	Defines the SS7 SUERM (Signal Unit Error Rate Monitor) T threshold. The valid range is 0 to 256. The default value is 64.
<b>SS7_MTP2_Param_Timer_T1</b>	Defines the SS7 MTP2 T1 alignment ready timer (in msecs). The valid range is 0 to 100000. The default value is 50000.
<b>SS7_MTP2_Param_Timer_T2</b>	Defines the SS7 MTP2 T2 unaligned timer (in msecs). The valid range is 0 to 200000. The default value is 150000.
<b>SS7_MTP2_Param_Timer_T3</b>	Defines the SS7 MTP2 T3 timer aligned. The valid range is 0 to 20000. The default value is 2000.
<b>SS7_MTP2_Param_Timer_T4E</b>	Defines the SS7 MTP2 T4e Emergency proving period timer (msec). The valid range is 0 to 5000. The default value is 500.
<b>SS7_MTP2_Param_Timer_T4N</b>	Defines the SS7 MTP2 T4n Nominal proving period timer. The valid range is 0 to 15000. The default value is 8200.
<b>SS7_MTP2_Param_Timer_T5</b>	Defines the SS7 MTP2 Sending SIB timer. The valid range is 0 to 2400. The default value is 120.

**Table J-1: SS7 Parameters (continues on pages 233 to 234)**

<i>ini</i> File Parameter Name	Description
<b>SS7_MTP2_Param_Timer_T6</b>	Defines the SS7 MTP2 Remote Congestion timer (in msec). The valid range is 0 to 10000. The default value is 6000.
<b>SS7_MTP2_Param_Timer_T7</b>	Defines the SS7 MTP2 excessive delay of the ack timer (in msec). The valid range is 0 to 5000. The default value is 2000.

## J.4 SS7 Table Parameters

### J.4.1 SIGTRAN Interface Groups

**Table J-2: SIGTRAN Interface Groups (continues on pages 234 to 235)**

<i>ini</i> File Parameter Name	Description
<b>SS7_SIG_IF_GR_INDEX</b>	Indicates the SS7 interface group index for a line. The valid range is 0 to 15.
<b>SS7_IF_GR_ID</b>	Determines the SS7 SIGTRAN interface group index, for a line. The valid range is 0 to 0xFFFF. The default value is 0xFFFE.
<b>SS7_SIG_SG_MGC</b>	Determines the SS7 SIGTRAN interface group Signaling Gateway (SG) and Media Gateway Controller (MGC) option. The valid range is 77(MGC), 83(SG). The default value is 83.
<b>SS7_SIG_LAYER</b>	Determines the SIGTRAN group layer (IUA/M2UA/M3UA). Choose either: 0 = no_layer (default) 1 = iua 2 = m2ua 3 = m3ua 4 = m2tunnel 5 = V5ua
<b>SS7_SIG_TRAF_MODE</b>	Determines the SS7 SIGTRAN interface group traffic mode. The valid range is 1 to 3. The default value is 1.
<b>SS7_SIG_T_REC</b>	Determines the SIGTRAN group T recovery. The valid range is 0 to 10000000. The default value is 2000.
<b>SS7_SIG_T_ACK</b>	Determines the SIGTRAN group T Ack (in msec). The valid range is 0 to 10000000. The default value is 2000.
<b>SS7_SIG_T_HB</b>	Determines the SIGTRAN group T Hb (in msec). The valid range is 0 to 10000000. The default value is 2000.
<b>SS7_SIG_MIN_ASP</b>	Determines the SIGTRAN group minimal Application Server Process (ASP) number (minimum = 1). The valid range is 1 to .10 The default value is 1.
<b>SS7_SIG_BEHAVIOUR</b>	Determines the SIGTRAN group behavior bit. The valid range is 0 to 0xFFFFFFFF. The default value is 0.
<b>SS7_SCTP_INSTANCE</b>	Determines the SIGTRAN group SCTP instance. The valid range is 0 to 0xFFFE. The default value is 0xFFFE.

Table J-2: SIGTRAN Interface Groups (continues on pages 234 to 235)

<i>ini</i> File Parameter Name	Description
<b>SS7_LOCAL_SCTP_PORT</b>	Determines the SIGTRAN group SCTP port. The valid range is 0 to 0xFFFFE. The default value is 0Xffe.
<b>SS7_SIG_NETWORK</b>	Determines the SIGTRAN group Network (ITU, ANSI, CHINA). The valid range is 1 to 3. The default value is 1.
<b>SS7_DEST_SCTP_PORT</b>	Determines the SIGTRAN group destination SCTP port. The valid range is 0 to 0xFFFFE. The default value is 0xFFFFE.
<b>SS7_DEST_IP</b>	Determines the SIGTRAN group destination IP Address The valid range is 0 to 0xFFFFFFFFE. The default value is 0.
<b>SS7_MGC_MX_IN_STREAM</b>	Determines the SIGTRAN group maximum inbound stream. The valid range is 2 to 0xFFFFE. The default value is 2.
<b>SS7_MGC_NUM_OUT_STREAM</b>	Determines the SIGTRAN group's number of outbound streams. The valid range is 2 to 0xFFFFE. The default value is 2.

## J.4.2 SIGTRAN Interface IDs

Table J-3: SIGTRAN Interface IDs

<i>ini</i> File Parameter Name	Description
<b>SS7_SIG_IF_ID_INDEX</b>	Determines the SS7 interface ID index, for a line. The valid range is 0 to 15. The default value is 1.
<b>SS7_SIG_IF_ID_VALUE</b>	Determines the SIGTRAN interface ID value. The valid range is 0 to 0xFFFFFFFFE. The default value is 0.
<b>SS7_SIG_IF_ID_NAME</b>	Determines the SIGTRAN interface ID (text string). The default string is 'INT_ID'.
<b>SS7_SIG_IF_ID_OWNER_GROUP</b>	Determines the SIGTRAN interface ID owner group. The valid range 0 to 0xFFFFE. The default value is 0.
<b>SS7_SIG_IF_ID_LAYER</b>	Determines the SIGTRAN group layer (IUA/M2UA/M3UA). 0 = no_layer (default) 1 = iua 2 = m2ua 3 = m3ua 4 = m2tunnel 5 = V5ua
<b>SS7_SIG_IF_ID_NAI</b>	Determines the SIGTRAN interface ID NAI. The valid range 0 to 0xFFFFE. The default value is 0xFFFFE.
<b>SS7_SIG_M3UA_SPC</b>	Determines the SIGTRAN M3UA SPC. The valid range 0 to 0xFFFFFFFFE. The default value is 0.

## J.4.3 SS7 Signaling Link

Table J-4: SS7 Signaling Link (continues on pages 236 to 237)

<i>ini</i> File Parameter Name	Description
<b>SS7_LINK_INDEX</b>	Determines the index field for a line. The valid range is 0 to 7. The default value is 0.
<b>SS7_LINK_ROWSTATUS</b>	Determines the RowStatus field for a line. The valid range is acPARAMSET_ROWSTATUS_DOESNOTEXIST to acPARAMSET_ROWSTATUS_DESTROY. The default value is acPARAMSET_ROWSTATUS_DOESNOTEXIST.
<b>SS7_LINK_ACTION</b>	Determines the management field for actions. The valid range is acSS7LINK_PS_ACTION_NONE to acSS7LINK_PS_ACTION_LPR. The default value is acSS7LINK_PS_ACTION_NONE.
<b>SS7_LINK_ACTION_RESULT</b>	Determines the management field for actions result. The valid range is acPARAMSET_ACTION_RESULT_SUCCEEDED to acPARAMSET_ACTION_RESULT_FAILED. The default value is acPARAMSET_ACTION_RESULT_SUCCEEDED.
<b>SS7_LINK_NAME</b>	String name for link parameters The default string is "LINK".
<b>SS7_LINK_OPERATIONAL_STATE</b>	Determines the operational state of a signaling link. The valid range is L3_OFFLINE to L3_INSERTSERVICE. The default value is L3_OFFLINE.
<b>SS7_LINK_ADMINISTRATIVE_STATE</b>	Determines the administrative state of a signaling link. The valid range is L3_OFFLINE to L3_INSERTSERVICE. The default value is L3_OFFLINE.
<b>SS7_LINK_TRACE_LEVEL</b>	Determines the trace level of a signaling link (level 2). The valid range is 0 to 1. The default value is 0.
<b>SS7_LINK_L2_TYPE</b>	Determines the link layer type - defines level 2 media of signaling link. The valid range is SS7_SUBLINK_L2_TYPE_NONE to SS7_SUBLINK_L2_TYPE_SAAL. The default value is SS7_SUBLINK_L2_TYPE_NONE.
<b>SS7_LINK_L3_TYPE</b>	Determines the link high layer type - defines level 3 or L2 high layer of signaling link. The valid range is SS7_SUBLINK_L3_TYPE_NONE to SS7_SUBLINK_L3_TYPE_MTP2_TUNNELING. The default value is SS7_SUBLINK_L3_TYPE_NONE.
<b>SS7_LINK_TRUNK_NUMBER</b>	Determines the trunk number of a signaling link (TDM). The valid range is 0 to 7. The default value is 0.
<b>SS7_LINK_TIMESLOT_NUMBER</b>	Determines the time-slot number of a signaling link (TDM). The valid range is 0 to 31. The default value is 16.
<b>SS7_LINK_MTC_BUSY</b>	Determines the link local busy indicator – if set, indicates link is busy due to local mtc action. The valid range is 0 to 1. The default value is 0.
<b>SS7_LINK_INHIBITION</b>	Determines the link inhibit indicator - if set, indicates link is inhibited. The valid range is 0 to 1. The default value is L3_LINK_UNINHIBITED.
<b>SS7_LINK_LAYER2_VARIANT</b>	Determines the variant (layer 2) of signaling link (TDM). The valid range is NET_VARIANT_OTHER to NET_VARIANT_CHINA. The default value is NET_VARIANT_ITU.
<b>SS7_LINK_MTP2_ATTRIBUTES</b>	Determines the MTP2 attributes of signaling link (TDM). The valid range is 0 to MAX_C7_MTP2_PARAMS_INDEX. The default value is 3.
<b>SS7_CONGESTION_LOW_MARK</b>	Determines the link congestion low mark of signaling link (TDM). The valid range is 0 to 255. The default value is 5.
<b>SS7_CONGESTION_HIGH_MARK</b>	Determines the link congestion high mark of signaling link (TDM). The valid range is 0 to 255. The default value is 20.
<b>SS7_LINK_M2UA_IF_ID</b>	Determines the interface ID (M2UA) of signaling link. The valid range is 0 to 0xFFFFFFFF. The default value is 0.

Table J-4: SS7 Signaling Link (continues on pages 236 to 237)

<i>ini</i> File Parameter Name	Description
<b>SS7_LINK_GROUP_ID</b>	Determines the group ID (M3UA) of signaling link. The valid range is 0 to 0xFFFF. The default value is 0.
<b>ATM_SAAL_LINK_PROFILE_NUM</b>	Determines the ATM SAAL Link profile number The valid range is 0 to (MAX_SAAL_PROFILES-1). The default value is 0.
<b>ATM_SAAL_LINK_TYPE</b>	Determines the ATM SAAL link Type PVC/SVC The valid range is ATM_VCC_TYPE_PVC to ATM_VCC_TYPE_SVC. The default value is ATM_VCC_TYPE_PVC.
<b>ATM_SAAL_LINK_PORT_NUM</b>	Determines the ATM SAAL link port num The valid range is 0 to ATMDB_ATM_MAX_INTERFACES_RANGE. The default value is 0.
<b>ATM_SAAL_LINK_VPI</b>	Determines the ATM SAAL link VPI The valid range is 0 to 255. The default value is 0.
<b>ATM_SAAL_LINK_VCI</b>	Determines the ATM SAAL link VCI The valid range is 0 to 0xFFFF. The default value is 0.
<b>SS7_LINK_TNL_MGC_LINK_NUMBER</b>	Determines the MTP2 Tunneling: MGC link number (MTP2 \other side\ of signaling link. The valid range is 0 to 7. The default value is 0.
<b>SS7_LINK_TNL_ALIGNMENT_MODE</b>	Determines the MTP2 Tunneling: Alignment mode of signaling links in tunnel. The valid range is 0 to 255. The default value is M3B_ALIGNMENT_EMERGENCY.
<b>SS7_LINK_TNL_CONGESTION_MODE</b>	Determines the MTP2 Tunneling: Congestion mode of signaling links in tunnel. The valid range is 0 to 255. The default value is M3B_CONGESTION_ACCEPT.
<b>SS7_LINK_TNL_WAIT_START_COMPLETE_TIMER</b>	Determines the MTP2 Tunneling Timer: wait start complete. The valid range is 500 to 0xFFFFFFFF. The default value is 30000.
<b>SS7_LINK_TNL_OOS_START_DELAY_TIMER</b>	Determines the MTP2 Tunneling Timer: OOS start delay. The valid range is 500 to 0xFFFFFFFF. The default value is 5000.
<b>SS7_LINK_TNL_WAIT_OTHER_SIDE_INSV_TIMER</b>	Determines the MTP2 Tunneling Timer: wait other side inservice. The valid range is 500 to 0xFFFFFFFF. The default value is 30000.
<b>SS7_LINKSET_SN_INDEX</b>	Determines the first index field for line. The valid range is 0 to 1. The default value is 0.
<b>SS7_LINKSET_LINKSET_INDEX</b>	Determines the second index field for line. The valid range is 0 to 7. The default value is 0.
<b>SS7_LINKSET_ROWSTATUS</b>	Determines the RowStatusField for line. The valid range is acPARAMSET_ROWSTATUS_DOESNOTEXIST to acPARAMSET_ROWSTATUS_DESTROY. The default value is acPARAMSET_ROWSTATUS_DOESNOTEXIST.
<b>SS7_LINKSET_ACTION</b>	Determines the management field for actions. The valid range is acSS7LINKSET_PS_ACTION_NONE to acSS7LINKSET_PS_ACTION_DEACTIVATE. The default value is acSS7LINKSET_PS_ACTION_NONE.
<b>SS7_SN_ACTION_RESULT</b>	Determines the management field for actions result. The valid range is acPARAMSET_ACTION_RESULT_SUCCEEDED to acPARAMSET_ACTION_RESULT_FAILED. The default value is acPARAMSET_ACTION_RESULT_SUCCEEDED.

## J.5 SS7 MTP2 Tunneling *ini* File Example

For the SS7 MTP2 tunneling *ini* file example, note the following:

- The first *ini* file acts as an MTP2 tunneling central side (M2UA MGC links).

- There are 8 SS7 links - 4 links of type: MTP2 MGC, and 4 links of type MTP2. Each pair of links (1 MTP2 MGC and 1 MTP2) defines an MTP2 tunnel.
- There is 1 interface that is used for the M2UA MGC <=> M2UA SG (Signaling Gateway) connection.
- There are 4 interface IDs defined: 1 per link (M2UA MGC side).
- This file is intended for ITU link variant (E1 trunks).

➤ **To load the example SS7 MTP2 tunneling *ini* files to Mediant 2000 gateways, take these 3 steps:**

1. Load the *ini* file that is shown in [Figure J-4](#) to a tunnel central gateway (MTP2 MGC). Load the *ini* file that is shown in [Figure J-5](#) to a tunnel remote gateway (MTP2 SG); the MGC gateway connects (over IP) to the SG gateway. For information on loading an *ini* file to the gateway, refer to [Section 6.2](#) on page 87.
2. In the MGC gateway, change the parameter 'SS7\_DEST\_IP' to the actual IP address of the M2UA SG gateway.
3. Change the value of the 'SyslogServerIP' parameter in the MGC and SG gateways to your Syslog server IP address.

**Figure J-4: SS7 MTP2 Tunneling *ini* File Example - MGC**

```
[TDM BUS configuration]

; l=aLaw 3=ulaw

PCMLawSelect= 1

;1 - internal, 3 - mvip, 4 - Network, 8 - h110a, 9 - h110b, 10 - Netref

TDMBusClockSource= 1

[Trunk Configuration]

;e1_euro_isdn=1 t1_isdn=2 ;e1_cas_r2=8 (8 for fcd); e1_trans_62=5

ProtocolType = 5

TraceLevel = 0

; acCLOCK_MASTER_ON =1

CLOCKMASTER= 1

;acUSER_TERMINATION_SIDE = 0

TerminationSide = 1

;acEXTENDED_SUPER_FRAME=0

FramingMethod = 0

;acB8ZS = 0 2 for E1_CAS - FCD

LineCode = 0

[SS7]

SS7_MTP2_PARAM_TIMER_T1_0=50000

SS7_MTP2_PARAM_TIMER_T2_0=150000

SS7_MTP2_PARAM_TIMER_T3_0=1000
```

**Figure J-4: SS7 MTP2 Tunneling *ini* File Example - MGC**

```

SS7_MTP2_PARAM_TIMER_T4E_0=500

SS7_MTP2_PARAM_TIMER_T4N_0=8200

SS7_MTP2_PARAM_TIMER_T5_0=100

SS7_MTP2_PARAM_TIMER_T6_0=3000

SS7_MTP2_PARAM_TIMER_T7_0=2000

[syslog]

SYSLOGSERVERIP = 168.100.0.1

ENABLESYSLOG = 1

;FORCEEXCEPTIONDUMP = 1

WATCHDOGSTATUS = 0

[ SS7_LINK_TABLE ]

FORMAT SS7_LINK_INDEX = SS7_LINK_NAME, SS7_LINK_TRACE_LEVEL,
SS7_LINK_ADMINISTRATIVE_STATE,SS7_LINK_L2_TYPE, SS7_LINK_L3_TYPE, SS7_LINK_GROUP_ID,
SS7_LINK_M2UA_IF_ID;

SS7_LINK_TABLE 1 = new_link_1, 0, 2, 2, 3, 4, 50;

SS7_LINK_TABLE 3 = new_link_3, 0, 2, 2, 3, 4, 12;

SS7_LINK_TABLE 5 = new_link_5, 0, 2, 2, 3, 4, 18;

SS7_LINK_TABLE 7 = new_link_7, 0, 2, 2, 3, 4, 1;

[ \SS7_LINK_TABLE ]

[ SS7_LINK_TABLE ]

FORMAT SS7_LINK_INDEX = SS7_LINK_NAME, SS7_LINK_TRACE_LEVEL,
SS7_LINK_ADMINISTRATIVE_STATE,SS7_LINK_L2_TYPE, SS7_LINK_L3_TYPE,
SS7_LINK_TRUNK_NUMBER,SS7_LINK_TIMESLOT_NUMBER,
SS7_LINK_LAYER2_VARIANT,SS7_LINK_MTP2_ATTRIBUTES,SS7_CONGESTION_LOW_MARK,
SS7_CONGESTION_HIGH_MARK, SS7_LINK_TNL_MGC_LINK_NUMBER, SS7_LINK_TNL_ALIGNMENT_MODE,
SS7_LINK_TNL_CONGESTION_MODE, SS7_LINK_TNL_WAIT_START_COMPLETE_TIMER,
SS7_LINK_TNL_OOS_START_DELAY_TIMER, SS7_LINK_TNL_WAIT_OTHER_SIDE_INSV_TIMER;

SS7_LINK_TABLE 0 = new_link_0, 0, 2, 1, 3, 0, 15, 1, 0, 5, 50, 1, 1, 0, 30000, 5000, 30000;

SS7_LINK_TABLE 2 = new_link_2, 0, 2, 1, 3, 3, 12, 1, 0, 5, 50, 3, 1, 0, 30000, 5000, 30000;

SS7_LINK_TABLE 4 = new_link_4, 0, 2, 1, 3, 6, 7, 1, 0, 5, 50, 5, 1, 0, 30000, 5000, 30000;

SS7_LINK_TABLE 6 = new_link_6, 0, 2, 1, 3, 7, 3, 1, 0, 5, 50, 7, 1, 0, 30000, 5000, 30000;

[ \SS7_LINK_TABLE ]

[ SS7_SIG_IF_GROUP_TABLE ]

FORMAT SS7_SIG_IF_GR_INDEX = SS7_IF_GR_ID,SS7_SIG_SG_MGC, SS7_SIG_LAYER, SS7_SIG_TRAF_MODE,
SS7_SIG_T_REC, SS7_SIG_T_ACK, SS7_SIG_T_HB, SS7_SIG_MIN_ASP, SS7_SIG_BEHAVIOUR,
SS7_LOCAL_SCTP_PORT, SS7_SIG_NETWORK, SS7_DEST_SCTP_PORT, SS7_DEST_IP, SS7_MGC_MX_IN_STREAM,
SS7_MGC_NUM_OUT_STREAM;

```

**Figure J-4: SS7 MTP2 Tunneling ini File Example - MGC**

```

SS7_SIG_IF_GROUP_TABLE 4 = 4, 77, 4, 1, 2000, 2000, 30000, 1, 0, 2904, 1,2904,168.100.0.2,3,3;

[ \SS7_SIG_IF_GROUP_TABLE ]

[ SS7_SIG_INT_ID_TABLE ]

FORMAT SS7_SIG_IF_ID_INDEX = SS7_SIG_IF_ID_VALUE, SS7_SIG_IF_ID_NAME, SS7_SIG_IF_ID_OWNER_GROUP,
SS7_SIG_IF_ID_LAYER, SS7_SIG_IF_ID_NAI, SS7_SIG_M3UA_SPC;

SS7_SIG_INT_ID_TABLE 7 = 50, BELFAST12, 4, 2, 1, 0;

SS7_SIG_INT_ID_TABLE 8 = 12, AMSTERDAM, 4, 2, 3, 0;

SS7_SIG_INT_ID_TABLE 9 = 18, ROTTERDAM , 4, 2, 5, 0;

SS7_SIG_INT_ID_TABLE 10 = 1, GAUDA , 4, 2, 7, 0;

[ \SS7_SIG_INT_ID_TABLE ]
    
```

**Figure J-5: SS7 MTP2 Tunneling ini File Example - SG**

```

[TDM BUS configuration]

; l=aLaw 3=ulaw

PCMLawSelect= 1

;1 - internal, 3 - mvip, 4 - Network, 8 - h110a, 9 - h110b, 10 - Netref

TDMBusClockSource= 1

[Trunk Configuration]

;e1_euro_isdn=1 t1_isdn=2 ;e1_cas_r2=8 (8 for fcd); e1_trans_62=5

ProtocolType = 5

TraceLevel = 0

; acCLOCK_MASTER_ON =1

ClockMaster= 1

TerminationSide = 1

;acEXTENDED_SUPER_FRAME=0

FramingMethod = 0

;acB8ZS = 0 2 for E1_CAS - FCD

LineCode = 0

WATCHDOGSTATUS = 0

[ SS7_LINK_TABLE ]

FORMAT SS7_LINK_INDEX = SS7_LINK_NAME, SS7_LINK_TRACE_LEVEL,
SS7_LINK_ADMINISTRATIVE_STATE,SS7_LINK_L2_TYPE, SS7_LINK_L3_TYPE,
SS7_LINK_TRUNK_NUMBER,SS7_LINK_TIMESLOT_NUMBER,SS7_LINK_M2UA_IF_ID;

SS7_LINK_TABLE 0 = new_link_0, 0, 2, 1,1, 1, 15,50;
    
```

**Figure J-5: SS7 MTP2 Tunneling ini File Example - SG**

```

SS7_LINK_TABLE 1 = new_link_1, 0, 2, 1,1, 2, 12, 12;

SS7_LINK_TABLE 2 = new_link_2, 0, 2, 1, 1, 4, 7,18;

SS7_LINK_TABLE 3 = new_link_3, 0, 2, 1, 1, 5, 3,1;

[\SS7_LINK_TABLE]

[ SS7_SIG_IF_GROUP_TABLE ]

FORMAT SS7_SIG_IF_GR_INDEX = SS7_IF_GR_ID,SS7_SIG_SG_MGC, SS7_SIG_LAYER, SS7_SIG_TRAF_MODE,
SS7_SIG_T_REC, SS7_SIG_T_ACK, SS7_SIG_T_HB, SS7_SIG_MIN_ASP, SS7_SIG_BEHAVIOUR,
SS7_LOCAL_SCTP_PORT, SS7_SIG_NETWORK;

SS7_SIG_IF_GROUP_TABLE 4 = 4,83, 2, 1, 2000, 2000, 30000, 1, 0, 2904, 1;

[ \SS7_SIG_IF_GROUP_TABLE ]

[ SS7_SIG_INT_ID_TABLE ]

FORMAT SS7_SIG_IF_ID_INDEX = SS7_SIG_IF_ID_VALUE, SS7_SIG_IF_ID_NAME, SS7_SIG_IF_ID_OWNER_GROUP,
SS7_SIG_IF_ID_LAYER, SS7_SIG_IF_ID_NAI, SS7_SIG_M3UA_SPC;

SS7_SIG_INT_ID_TABLE 7 = 50, BELFAST12, 4, 2, 0, 0;

SS7_SIG_INT_ID_TABLE 8 = 12, AMSTERDAM, 4, 2, 1, 0;

SS7_SIG_INT_ID_TABLE 9 = 18, ROTTERDAM , 4, 2, 2, 0;

SS7_SIG_INT_ID_TABLE 10 = 1, GAUDA , 4, 2, 3, 0;

[ \SS7_SIG_INT_ID_TABLE ]

```

## J.6 *ini* File Parameters in a Table Format

Tables of Parameter Values group related parameters of a specific entity and handle them together. Tables are composed of lines and columns. The columns represent parameters types, while each line represents an entity. The parameters in each line are called the line attributes. Lines in table may represent (for example) a trunk, SS7 Link, list of timers for a given application, etc.

Table J-5 and Table J-6 below provide useful examples for reference.



**Note:** Table J-5 and Table J-6 are provided as examples for the purpose of illustration only and are NOT actually implemented in AudioCodes products.

**Table J-5: Table of Parameter Values Example - Remote Management Connections**

Index Fields: 1. Connection Number				
Connection Number	User Name	User Password	Time Connected (msec)	Permissions
0	Admin	Yellow9	0	All
1	Gillian	Red5	1266656	Read Only
2	David	Orange6	0	Read Write

**Table J-6: Table of Parameter Values Example - Port-to-Port Connections**

Index Fields: 1. Source Ports 2. Destination IP 3. Destination Port				
Source Port	Destination IP	Destination Port	Connection Name	Application Type
2020	10.4.1.50	2020	ATM_TEST_EQ	LAB_EQ
2314	212.199.201.20	4050	ATM_ITROP_LOOP	LAB_EQ
6010	10.3.3.41	6010	REMOTE_MGMT	MGMT

## J.6.1 Table Indices

Each line in a table must be unique. For this reason, each table defines one or more Index fields. The combination of the Index fields determines the 'line-tag'. Each line-tag may appear only once.

In the example provided in [Table J-5](#), there is only one index field. This is the simplest way to mark lines.

In the example provided in [Table J-6](#), there are three Index fields. This more complicated method is a result of the application it represents.

## J.6.2 Table Permissions

Each field in a line has a 'permission' attribute, which determines if and when the user may modify the field.

There are several types of permissions:

- Read - The user may read the value of a field (true for all fields).
- Write - The user may modify the value of the field at any time.
- Create - The user must provide a value for the field at creation time.

The default values set for all fields already determine the initial values.

- Maintenance write - The user may modify the value of the field only when the entity represented by the line is in maintenance state.

Each table includes rules to determine when it is in a maintenance state.

In the example in [Table J-5](#), 'User Name' and 'User Password' fields have Read-Create permissions. The 'Time Connected' field has Read-Only permission, and the 'Permissions' field has a Read-Create-Maintenance\_write permission.

## J.6.3 Tables of Parameter Value Rules in the *ini* File Structure

The *ini* file allows you to add/modify parameters in tables. When using tables, Read-Only parameters are not loaded, since they cause an error when trying to download the loaded file. Therefore read-only parameters should not be included in tables in the *ini* file. Consequently, tables are loaded with all parameters having at least one of the following permissions:

- Write
- Create
- Maintenance write

The 'format-line' rule defines which fields of the table are to be modified by the given *ini* file (this may vary among *ini* files for the same table). The 'format-line' must only include fields, which can be modified (which are all parameters that are not specified as read-only).

One exception is the index-fields, which are ALWAYS mandatory fields. In the example provided in Table J-5, all fields except the 'Time Connected' field are loaded.

### J.6.3.1 Tables Structure Rules

Tables are composed of four elements:

- Table-Title - The Table's string name in square brackets (e.g. [ MY\_TABLE\_NAME ]).
- Format Line - This line specifies the table's fields by their string names.
  - The first word MUST be "FORMAT", followed by indices field names, and after '=' sign, all data fields names should be listed.
  - Items must be separated by ',' sign.
  - The Format Line must end with ';' sign.
- Data Line(s) - The actual values for parameters are specified in each Data line. The values are interpreted according to the format line. The first word must be the table's string name.
  - Items must be separated by a ',' sign.
  - A Data Line must end with a ';' sign.
- End-of-Table-Mark: Marks the end of a table. Same as Table title, but string name is preceded by '\'.

Figure J-6 displays an example of the Table structure in an *ini* file.

**Figure J-6: Structure of a Table in an ini File**

```

; Table: Items Table.
; Fields: Item_Name, Item_Serial_Number, Item_Color, Item_weight.
; NOTE: Item_Color is not specified. It will be given default value.
[Items_Table]
; Fields declaration
Format Item_Index = Item_Name, Item_Serial_Number, Item_weight;
Items_Table 0 = Computer, 678678, 6;
Items_Table 6 = Computer-screen, 127979, 9;
Items_Table 2 = Computer-pad, 111111, $$;
[\Items_Table]

```

- Indices (in both the Format line and the Data lines) must all appear in order, as determined by the table's specific documentation. The Index field must NOT be omitted.
- Data fields in the Format line may use a sub-set of all of the configurable fields in a table only. In this case, all other fields are assigned with the pre-defined default value for each configured line.

- The order of the Data fields in the Format line is not significant (unlike the Index-fields). Field values in Data lines are interpreted according to the order specified in the Format line.
- The sign '\$\$' in the Data line means that the user wants the pre-defined default value assigned to the field for the given line.
- The order of Data lines is insignificant.
- Data lines must match the Format line, i.e., it must contain exactly the same number of Indices and Data fields and should be in exactly the same order.
- A line in a table is identified by its table-name and its indices. Each such line may appear only once in the *ini* file.
- Tables' dependencies:

Certain tables may depend on other tables. For example, one table may include a field, which specifies an entry in another table, to specify additional attributes of an entity, or to specify that a given entity is part of a larger entity. The tables must appear in order of dependency (i.e., if Table X is referred to by Table Y, then Table X must appear in the *ini* file before Table Y).

### J.6.3.2 Dynamic Tables versus Static Tables

Static Table:

The Static table type does not support adding new lines or removing (deleting) an existing line. All lines in a Static table are pre-configured with default values. Users may modify values in existing lines. After reset, all lines in a Static table are available.

Dynamic Table:

The Dynamic table type supports adding and removing lines. It is always initialized as an empty table, with no lines. Users should add lines to the Dynamic table via the *ini* file or at run-time.



**Note:** Certain dynamic tables may initialize a line (or more) at start-up time. If so, it is documented in the table's specific section.

### J.6.3.3 Tables in the Loaded *ini* File

Tables are grouped according to the applications they configure. For example, several tables are required to configure SS7, and other tables are required to configure ATM.

When loading the *ini* file, the policy is to include only tables that belong to applications, which have been configured (Dynamic tables of other applications are empty, but static tables are not).

## Appendix K RADIUS Billing and VXML Calling Card Application

The Mediant 2000 calling card application capability (included in its IVR - Interactive Voice Response - feature) enables Internet Telephony Service Providers (ITSPs) to provide a VoIP telephone service to subscribers who have purchased calling cards in advance.

The subscriber market for calling cards is growing exponentially worldwide. Calling cards are often much cheaper than collect calls and operator-assisted calls made through long distance providers and local phone companies. VoIP service providers, using the Mediant 2000, can *further reduce costs* for calling card subscribers and substantially reduce implementation time, making the service extraordinarily attractive both for them and their subscribers.

### K.1 Benefits

Using the Mediant 2000, telephony service providers can offer the calling card service over a VoIP network and thereby:

- Lower the cost and deployment time that a PSTN calling-card service requires
- Achieve voice quality comparable to toll quality
- Acquire a cost-effective, reliable VoIP network infrastructure
- IVR functionality can be located at the edge of the network (distributed functionality) or in a central location
- Connect with the PSTN over carrier interfaces
- Interoperate with other VoIP service providers and other vendors' VoIP equipment
- Become part of a world-wide network of other VoIP service providers interested in interconnecting

### K.2 Features

- PSTN→IP and IP→PSTN distributed IVR architecture
- AAA (Authentication, Authorization, Accounting) over standard RADIUS (Remote Authentication Dial In User Service) - stored on a RADIUS server
- Provides comprehensive management of accounting and billing support
- CDRs (Call Detail Reports) over RADIUS (stored on a RADIUS server)
- Post-paid applications (Billing model: credit)
- Pre-paid application (Billing model: debit). When credit is exhausted the call is disconnected (a short Prompt is played prior to disconnection)
- Common internal, on-board, Voice Prompts (in flash memory) for all VoiceXML (Voice Extensible Markup Language) scripts
- Multiple VXML scripts stored on external HTTP server (up to 10 different scripts)
- Caller can place multiple successive calls without re-entering the account and password numbers (authentication and authorization are applied without collecting the information from the user again)
- Supports Cisco gateway RADIUS functionality
- Interoperates with standard RADIUS-based (AAA) billing servers
- Supports 240 concurrent calls running a VXML script

- Loads the VXML scripts once and stores them in the RAM; scripts can be changed without the need of reset
- Barge-in dialing (to shorten menu time), once prompt has started

### K.3 Supported Architecture

Figure K-1: Mediant 2000 Supported Architecture

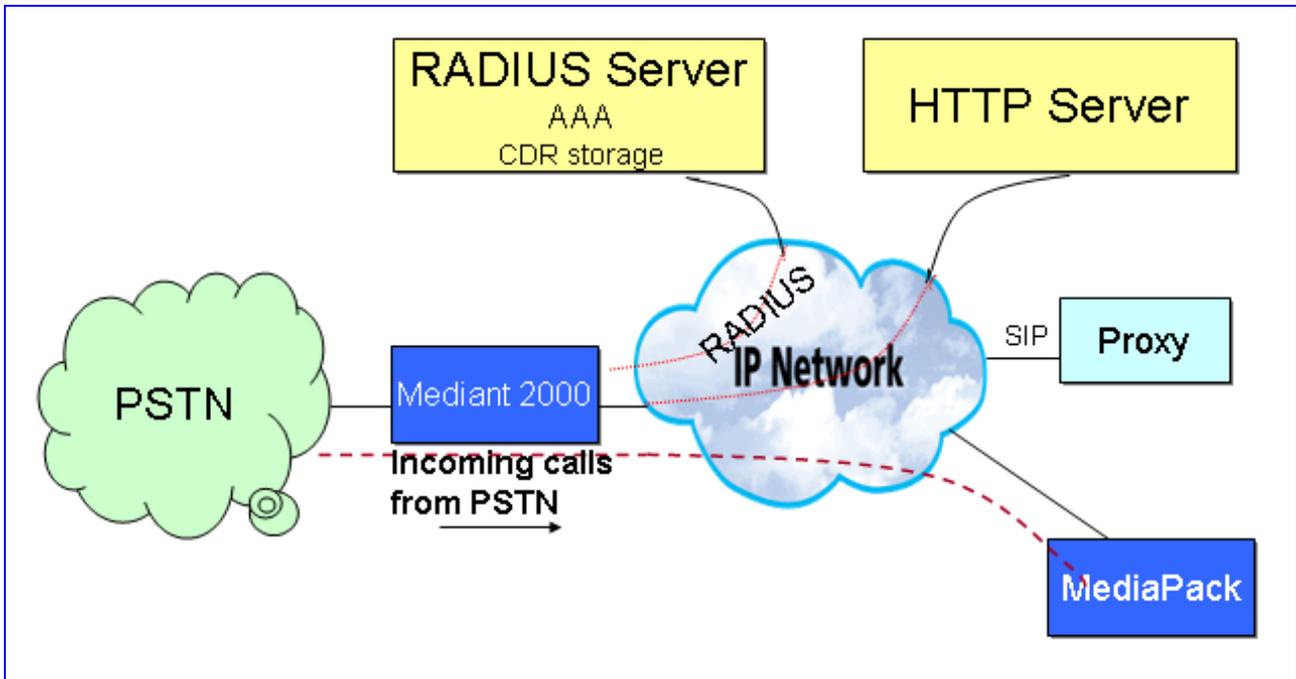


Figure K-1 illustrates standard Calling Card IVR application architecture. The figure depicts in general terms an incoming PSTN→IP call being conveyed to the IP network.

The architecture comprises the following components:

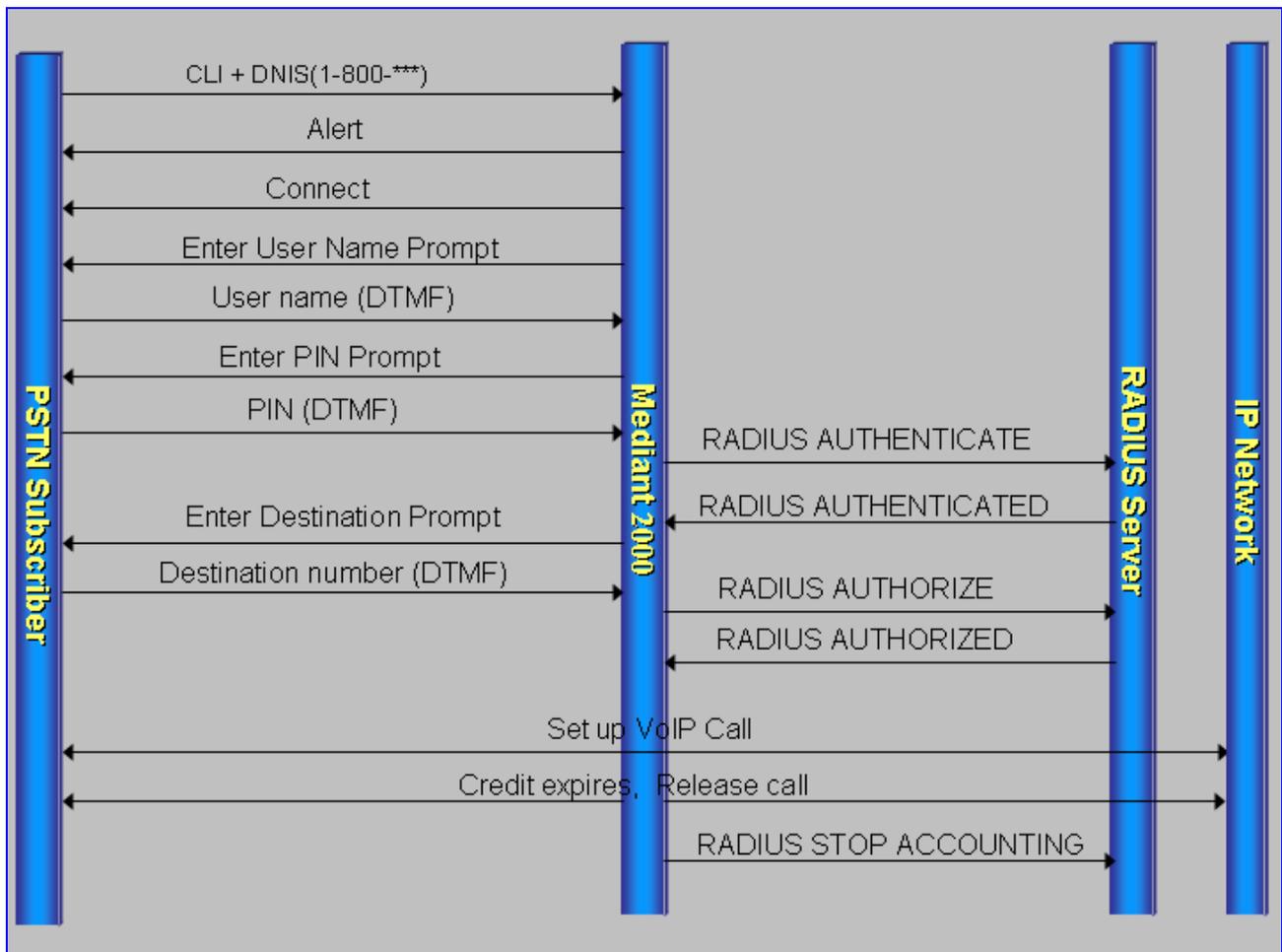
- **Mediant 2000** - The PSTN gateway that includes the VoiceXML interpreter which generates events in response to user actions (e.g., spoken or character input received, disconnect) and system events (e.g., timer expiration). These events are acted on by the VoiceXML interpreter itself, as specified by the VoiceXML document.
- **HTTP Server** – (Document Server), sends out the VoiceXML script in response to the Mediant 2000 (the VoiceXML interpreter) request.
- **RADIUS Server** – A centralized Authentication, Authorization and Accounting server for remote access users that communicate via the RADIUS protocol.
- **VoiceBrowser** - (not shown in this diagram), responsible for the TTS (Text to Speech) and ASR (Automatic Speech Recognition) services (*not supported in this version*).
- **Proxy Server** – Standard management tool for SIP networks. Performs essential control, administrative and managerial functions.
- **MediaPack** – Analog media gateway, provides excellent voice quality and optimized packet voice, fax and modem streaming over the IP network.

## K.4 Implementation

The Mediant 2000 uses an embedded VoiceXML interpreter to interpret and execute standard VoiceXML scripts, which are loaded from an outbound HTTP server and stored in the gateway's volatile memory (RAM). The predefined VoiceXML scripts (up to 10 different scripts are supported) determine the development of the call according to the caller's responses (DTMF digits) and AAA (Authentication, Authorization and Accounting) information exchanged with a RADIUS server. Interaction with the caller is conducted using a set of audio messages (stored in a single Voice Prompts file in the gateway's flash/RAM memory) on the gateway's side, and by pressing DTMF digits on the subscriber's side.

### K.4.1 Basic Calling Card IVR Scenario

Figure K-2: Basic Call Scenario



**Note:** Some messages have been omitted from the above drawing for the sake of clarity.

## K.4.2 Call Flow Description

Figure K-2 on the previous page depicts an example of a standard PSTN→IP call (billing-model: debit).

- An incoming PSTN call with a published access number reaches the Mediant 2000.
- The Mediant 2000 accepts the call (sends an Alert message).
- The Mediant 2000 searches its internal Tel→IP Destination Number Manipulation table for the specific prefix. When it is detected, if the 'Prefix to Add' column corresponds to the predefined VXMLID parameter (in the example below: *http*), the Mediant 2000 determines that the incoming call is an IVR call.

**Figure K-3: Basic *ini* File VXML Parameters**

```
VXMLID = http
NumberMapTel2IP = 5394288,7,http://10.8.1.19/RadAAA01.txt
```

- The Mediant 2000 loads the VoiceXML file (*RadAAA01.txt*) from an outbound HTTP Server (*10.8.1.19*) and stores it in its volatile memory. In Figure K-2 it is assumed that the VoiceXML already resides in the Mediant 2000.
- The Mediant 2000 sends an Answer/Connect message.



**Note:** The following steps are according to the supplied VXML script.

- The Mediant 2000 starts by playing an initial voice message. This message is composed of an opening greeting and an interactive menu asking the caller to choose one of the following options: 1 to make a call, 2 for help, 3 for operator service, 4 to exit.
- After pressing the digit 1, the caller is immediately prompted to enter his account number (usually the card number) and his password or PIN (personal identification number). The Mediant 2000 collects the input DTMF digits and sends an *Authentication* message to an outbound RADIUS server.
- Only after the call is authenticated successfully (in this stage the RADIUS server returns the billing-model, in the above example: debit), the caller is asked to enter the number he wishes to reach, the final destination number. The gateway collects the input DTMF digits and sends an *Authorization* message to the RADIUS server.
- The RADIUS server determines if the caller is authorized to proceed with the call and specifies the maximum duration of the call; the Mediant 2000 conveys the call to the IP network and starts an internal timer.
- One minute before credit is exhausted (refer to the VXML parameter [finalalerttime](#) on page 260); the [finalalertaudio](#) is played; finally, a minute later, the call is disconnected and the [endaudio](#) Voice Prompt is played.
- After the conclusion of the call, the Mediant 2000 sends an *Accounting* message to the RADIUS server containing the call details (CDR) and prompts the user either to proceed with another call or to disconnect.

## K.5 Operation & Configuration

➤ **To start working with the IVR system, take these 6 steps:**

1. Install the Mediant 2000 (refer to Section [Installing the Mediant 2000](#) on page 27 and to Section [Getting Started](#) on page 35).
2. Create and load a Voice Prompts file to the Mediant 2000 (refer to Section 7.3 on page 137).
3. Create the VXML scripts (refer to Section [K.9](#) on page 254).
4. Install an HTTP Server, store the VXML scripts in it and provision the Mediant 2000 relevant VoiceXML parameters.
5. In the Mediant 2000 Destination Manipulation tables, (PSTN→IP and IP→PSTN) convert the predefined number to the HTTP server IP address and the VXML file name.

Assuming that the incoming number is 5394288, the IP address of the HTTP server is 10.8.1.19 and the name of the VXML file is RadAAA01.txt, then the *ini* file entry should look like: NumberMapTel2IP = 5394288,7,http://10.8.1.19/RadAAA01.txt. For the Mediant 2000 to identify that the incoming call is designated to the IVR system, assign the string "http" to the VXMLID parameter (refer to [Table K-1](#) on page 249 for extended provisioning information).



**Note:** To change a VXML script on-the-fly, access the Destination Number Manipulation table via the Embedded Web Server and replace the name of the file with the name of the new script (existing calls continue to run on the old script, while new calls run on the new script).

6. Install a RADIUS server, define the Mediant 2000 in it (specify a common 'SharedSecret' parameter) and configure the Mediant 2000 relevant RADIUS parameters (refer to [K.9](#) on page 254).

You are now ready to start using the Calling Card Application.

## K.6 Configuration Parameters

**Table K-1: General Mediant 2000 Parameters**

<i>ini</i> File Field Name	Valid Range and Description	IVR Reference
<b>NumberMapTel2IP</b>	<p>Manipulates the destination number for Tel to IP calls. NumberMapTel2IP = a,b,c,d,e,f,g</p> <p>a = Destination number prefix                      b = Number of stripped digits from the left, or (if brackets are used) from the right. A combination of both options is allowed.                      c = String to add as prefix, or (if brackets are used) as suffix. A combination of both options is allowed.                      d = Number of remaining digits from the right                      e = Number Plan used in RPID header                      f = Number Type used in RPID header                      g = Source number prefix</p> <p>The 'b' to 'f' manipulations rules are applied if the called and calling numbers match the 'a' and 'g' conditions.</p> <p>The manipulation rules are executed in the following order: 'b', 'd' and 'c'.                      Parameters can be skipped by using the sign "\$\$", for example:                      NumberMapTel2IP=2222,4,http://10.3.0.2/AAIP.txt</p>	<p>Replaces the incoming destination number with a URL that indicates where the VXML script is located.</p> <p>a = Calling Card number                      b = The length of the Calling Card number                      c = http://&lt;HTTP server IP&gt; + / + &lt;name of VXML file&gt;                      (maximum length of string is 50 characters)</p> <p><b>Note:</b> To update the VoiceXML file, change the name of the file in the 'c' field.</p>

<i>ini</i> File Field Name	Valid Range and Description	IVR Reference
<b>NumberMapIP2Tel</b>	<p>Manipulate the destination number for IP to Tel calls. NumberMapIP2Tel = a,b,c,d,e,f,g,h,i</p> <p>a = Destination number prefix b = Number of stripped digits from the left, or (if brackets are used) from the right. A combination of both options is allowed. c = String to add as prefix, or (if brackets are used) as suffix. A combination of both options is allowed. d = Number of remaining digits from the right e = Q.931 Number Plan f = Q.931 Number Type g = Source number prefix h = Not applicable, set to \$\$ i = Source IP address</p> <p>The 'b' to 'f' manipulation rules are applied if the called and calling numbers match the 'a', 'g' and 'i' conditions.</p> <p>The manipulation rules are executed in the following order: 'b', 'd' and 'c'. Parameters can be skipped by using the sign "\$\$", for example: NumberMapIP2Tel =2222,4,http://10.3.0.2/AAIP.txt <b>Note:</b> The Source IP address can include the "x" wildcard to represent <u>single</u> digits. For example: 10.8.8.xx represents all the addresses between 10.8.8.10 to 10.8.8.99.</p>	<p>Replaces the incoming destination number with a URL that indicates where the VXML script is located.</p> <p>a = Calling Card number b = The length of the Calling Card number c = http://&lt;HTTP server IP&gt; + / + &lt;name of VXML file&gt; (maximum length of string is 50 characters)</p> <p><b>Note:</b> To update the VoiceXML file, change the name of the file in the 'c' field.</p>
<b>VoicePromptsFileName</b>	The name (and path) of the file containing the Voice Prompts definitions.	
<b>SaveConfiguration</b>	Set to 1 to store the Voice Prompts file in the non-volatile memory (file size mustn't exceed 1Mb).	
<b>EnableVoiceStreaming</b>	0 = Disable voice streaming (default). 1 = Enable voice streaming.	Set to 1 to enable the load of the VoiceXML file from the HTTP server.

**Table K-2: VoiceXML Related Parameters**

<i>ini</i> File Field Name	Valid Range and Description
<b>EnableVxml</b> [Enable VXML]	0 = Disable the VXML feature (default). 1 = Enable the VXML feature.
<b>VxmlID</b> [VXML ID]	According to this string, the Mediant 2000 recognizes that an incoming call is to be diverted to the IVR system. <b>Note:</b> Set to "http" (the "http" string must also appear in the manipulation table).
<b>VxmlCollectDigits</b>	Determines the destination to which the VXML script reports the collected number (username).  0 = The collected number (username) is sent for authentication to the RADIUS server (default). 1 = The collected number is sent (in INFO message) to an Application / Proxy server.

The following RADIUS related parameters are described in [Table 6-1](#) on page 90:

- EnableRADIUS
- MaxRADIUSSessions
- SharedSecret
- RADIUSRetransmission
- RADIUSTo

- RADIUSAuthServerIP
- RADIUSAuthPort
- RADIUSAccServerIP
- RADIUSAccPort
- AAAIndications
- RADIUSAccountingType

## K.7 Supported RADIUS Attributes

Use Table K-3 below for explanations on the RADIUS attributes contained in the communication packets transmitted between the Mediant 2000 and a RADIUS Server.

**Table K-3: Supported RADIUS Attributes (continues on pages 251 to 252)**

Attribute Number	Attribute Name	VSA No.	Purpose	Value Format	Sample	AAA <sup>2</sup>
<b>Request Attributes</b>						
1	User-Name		Account number or calling party number or blank	String up to 15 digits long	5421385747	Start Acc Stop Acc Authe Autho
2	User-Password		User password	Up to 15 digits		Autho
4	NAS-IP-Address		IP address of the requesting Mediant 2000	Numeric	192.168.14.4 3	Start Acc Stop Acc Authe Autho
6	Service-Type		Type of service requested	Numeric	1: login	Start Acc Stop Acc
26	h323-incoming-conf-id	1	H.323/SIP call identifier	Up to 32 octets		Start Acc Stop Acc
26	h323-remote-address	23	IP address of the remote gateway	Numeric		Stop Acc
26	h323-conf-id	24	H.323/SIP call identifier	Up to 32 octets		Start Acc Stop Acc Authe Autho
26	h323-setup-time	25	Setup time in NTP format 1	String		Start Acc Stop Acc
26	h323-call-origin	26	The call's originator: Answering (IP) or Originator (PSTN)	String	Answer, Originate etc	Start Acc Stop Acc
26	h323-call-type	27	Protocol type or family used on this leg of the call	String	VoIP	Start Acc Stop Acc
26	h323-connect-time	28	Connect time in NTP format	String		Stop Acc
26	h323-disconnect-time	29	Disconnect time in NTP format	String		Stop Acc
26	h323-disconnect-cause	30	Q.931 disconnect cause code	Numeric		Stop Acc

<sup>2</sup> The values in column 'AAA' are as follows:

- 'Start Acc' - Start Accounting
- 'Stop Acc' - Stop Accounting
- 'Authe' - Authentication
- 'Autho' - Authorization

**Table K-3: Supported RADIUS Attributes (continues on pages 251 to 252)**

Attribute Number	Attribute Name	VSA No.	Purpose	Value Format	Sample	AAA <sup>2</sup>
26	h323-gw-id	33	Name of the gateway	String	SIPIDString	Start Acc Stop Acc
30	Called-Station-Id			String	8004567145	Start Acc
			Destination phone number	String	2427456425	Stop Acc Autho
31	Calling-Station-Id		Calling Party Number (ANI)	String	5135672127	Start Acc Stop Acc Authe Autho
40	Acct-Status-Type		Account Request Type (start or stop)	Numeric	1: start, 2: stop	Start Acc Stop Acc
41	Acct-Delay-Time		No. of seconds tried in sending a particular record	Numeric	5	Start Acc Stop Acc
42	Acct-Input-Octets		Number of octets received for that call duration	Numeric		Stop Acc
43	Acct-Output-Octets		Number of octets sent for that call duration	Numeric		Stop Acc
44	Acct-Session-Id		A unique accounting identifier - match start & stop	String	34832	Start Acc Stop Acc
46	Acct-Session-Time		For how many seconds the user received the service	Numeric		Stop Acc
47	Acct-Input-Packets		Number of packets received during the call	Numeric		Stop Acc
48	Acct-Output-Packets		Number of packets sent during the call	Numeric		Stop Acc
61	NAS-Port-Type		Mediant 2000 physical port type on which the call is active	String	0: Asynchronou s	Start Acc Stop Acc Authe Autho
<b>Response Attributes</b>						
26	h323-credit-time	102	Number of seconds for which the call is authorized	Numeric	360	Autho
26	h323-return-code	103	The reason for failing authentication (0 = ok, other number failed)	Numeric	0 Request accepted	Authe Autho Stop Acc
26	h323-billing-model	109	Type of billing service for a specific call	Numeric	1:debit/prepaid	Authe
44	Acct-Session-Id		A unique accounting identifier – match start & stop	String		Stop Acc

## K.8 RADIUS Server Messages



**Note:** In Figure K-4, Figure K-5 and Figure K-6 non-standard parameters are preceded with brackets.

### K.8.1 Authentication

**Figure K-4: Authentication Example**

```
Access-Request (116)
user-name = 111
user-password = (encrypted)
nas-ip-address = 212.179.22.213
nas-port-type = 0
calling-station-id = 202
// Authentication non-standard parameters:
(4923 24) h323-conf-id = 02102944 600a1899 3fd61009 0e2f3cc5
```

In the Access-Accept response, the RADIUS server sends the billing model:

```
(4923 109) h323-billing-model = 1/0
```

The billing model is a non-standard parameter and can be one of the following:

- 1 = credit (prepaid)
- 0 = debit (postpaid)

When a billing model isn't received, the Mediant 2000 assumes a prepaid billing model (1).

### 12.1.1 Authorization

**Figure K-5: Authorization Example**

```
Access-Request (121)
user-name = 111
user-password = (encrypted)
nas-ip-address = 212.179.22.213
nas-port-type = 0
called-station-id = 201
calling-station-id = 202
// Authorization non-standard parameters:
(4923 24) h323-conf-id = 02102944 600a1899 3fd61009 0e2f3cc5
```

In the Access-Accept response, the RADIUS server sends the credit time:

```
(4923 102) h323-credit-time = 6000
```

The credit-time is a non-standard parameter which is measured in seconds.

## 12.1.2 Accounting

**Figure K-6: Accounting Example**

```

Accounting-Request (361)
user-name = 111
acct-session-id = 1
nas-ip-address = 212.179.22.213
nas-port-type = 0
acct-status-type = 2
acct-input-octets = 4841
acct-output-octets = 8800
acct-session-time = 1
acct-input-packets = 122
acct-output-packets = 220
called-station-id = 201
calling-station-id = 202
// Accounting non-standard parameters:
(4923 33) h323-gw-id =
(4923 23) h323-remote-address = 212.179.22.214
(4923 1) h323-ivr-out = h323-incoming-conf-id:02102944 600a1899
3fd61009 0e2f3cc5
(4923 30) h323-disconnect-cause = 22 (0x16)
(4923 27) h323-call-type = VOIP
(4923 26) h323-call-origin = Originate
(4923 24) h323-conf-id = 02102944 600a1899 3fd61009 0e2f3cc5
    
```

## K.9 Voice XML Interpreter

VoiceXML (Voice Extensible Markup Language) is designed for creating audio dialogs that feature synthesized speech, digitized audio, recognition of speech and DTMF inputs, recording of spoken input, telephony, and mixed initiative conversations. Its major goal is to bring the advantages of Web-based development and content delivery to interactive voice response applications.

### K.9.1 Features

- Supports DTMF recognition.
- Executes audio dialogs between the gateway and a user, supporting mixed initiative applications.
- Audio prompt recording (currently not supported).
- Transfer Support - Using the <Transfer> element, the Mediant 2000 places a call to different destinations.
- JavaScript Expression Support - Supports ECMA script specification 3.0 (standard ECMA-262).
- Supports definition of the end-dial key ('\*' or '#') that terminates the DTMF collection. To define whether \* or # are used to terminate the DTMF collection, add the following line to each script:  
 <property name="EndDialKey" value="\*"/>  
 OR  
 <property name="EndDialKey" value="#"/>
- Supports number concatenation, enabling number modification per VXML script.

```

<filled>
  <assign name="user_passwd" expr="'domain'+user_passwd + '.com'"/>
    <return namelist="user_account_num user_passwd"/>
</filled>
    
```

The user\_passwd parameter (that initially contained the user password collected from the user) is being assigned the value 'domain'+user\_passwd + '.com'.

- Calling number (received from SIP incoming call) can optionally be used for authentication instead of the user name.

```
form id="GetCallerId">
<log label="VXML--> getting caller id from SIP..." />
<object name="FCallerId" classid="builtin://com.audiocodes.ulp.input">
<filled>
<if cond="FCallerId.Result != 'fail'">
<assign name="CallerId" expr="FCallerId.Result" />
<goto next="#PerformAuthenWithoutUserName" />
<else />
<goto next="#PerformAuthen" />
</if>
</filled>
</object>
</form>
```

## K.10 Supported Elements & Attributes

Table K-4: VoiceXML Supported Elements & Attributes (continues on pages 256 to 260)

Element	Element's Description	Parameters	Parameter's Description	Supported
<b>&lt;assign&gt;</b>	Assign value to variable	<i>name</i>	The name of the modified variable	✓
		<i>expr</i>	The new value of the variable.	
<b>&lt;audio&gt;</b>	Plays an audio clip within a prompt	<i>expr</i>	Dynamically determines the URI to fetch by evaluating this ECMAScript expression.	✓
<b>&lt;block&gt;</b>	A container of (non-interactive) executable code			✓
<b>&lt;catch&gt;</b>	Catch an event	<i>event</i>	The event or events to catch.	✓
		<i>count</i>	The occurrence of the event (default 1).	
		<i>cond</i>	An expression which must evaluate to true after conversion to Boolean in order for the event to be caught, (defaults true).	
<b>&lt;choice&gt;</b>	Defines a menu item	<i>dtmf</i>	The DTMF sequence for this choice.	✓
		<i>accept</i>	Override the setting for accept in <menu> for this particular choice.	
		<i>next</i>	URI of the next dialog or doc.	
		<i>expr</i>	Specifies an expression to evaluate as a URI to transition to instead of specifying a next.	
		<i>event</i>	Specify an event to be thrown instead of specifying a next.	
		<i>eventexpr</i>	An ECMAScript expression evaluating to the name of the event to be thrown.	
		<i>message</i>	A message string providing additional context about the event being thrown.	
		<i>messageexpr</i>	An ECMAScript expression evaluating to the message string.	
<b>&lt;clear&gt;</b>	Clear one or more form item variables	<i>namelist</i>	The list of variables to be reset.	✓
<b>&lt;disconnect&gt;</b>	Disconnect a session			✓
<b>&lt;else&gt;</b>	Used in <if> elements			✓
<b>&lt;elseif&gt;</b>	Used in <if> elements			✓
<b>&lt;error&gt;</b>	Catch an error event	<i>count</i>	The event count	✓
		<i>cond</i>	An optional condition to test to see if the event is caught by this element. Defaults to true.	
<b>&lt;exit&gt;</b>	Exit a session	<i>expr</i>	A return expression.	✓
		<i>namelist</i>	Variable names to be returned to interpreter context.	
<b>&lt;field&gt;</b>	Declares an input field in a form	<i>name</i>	The form item variable in the dialog scope that holds the result.	✓
		<i>expr</i>	The initial value of the form item variable;	
		<i>cond</i>	An expression that must evaluate to true after conversion to Boolean in order for the form item to be visited.	
		<i>type</i>	The type of field.	
		<i>slot</i>	The name of the grammar slot used to populate the variable. <b>(Not Supported)</b>	

**Table K-4: VoiceXML Supported Elements & Attributes (continues on pages 256 to 260)**

Element	Element's Description	Parameters	Parameter's Description	Supported
		<i>modal</i>	If this is false (the default) all active grammars are turned on while collecting this field.	
<b>&lt;filled&gt;</b>	An action executed when fields are filled	<i>mode</i>	Either all (the default), or any.	✓
		<i>namelist</i>	The input items to trigger on.	
<b>&lt;form&gt;</b>	A dialog for presenting information and collecting data	<i>id</i>	The name of the form.	✓
		<i>scope</i>	The default scope of the form's grammars.	
<b>&lt;goto&gt;</b>	Go to another dialog in the same or different document	<i>next</i>	The URI to which to transition.	✓
		<i>expr</i>	An ECMAScript expression that yields the URI.	
		<i>nextitem</i>	The name of the next form item to visit in the current form.	
<b>&lt;grammar&gt;</b>	Specify a speech recognition or DTMF grammar	<i>version</i>	Defines the version of the grammar. <b>(Not Supported)</b>	✓
		<i>xml:lang</i>	The language identifier of the contained or referenced grammar. <b>(Not Supported)</b>	
		<i>mode</i>	Defines the mode of the contained or referenced grammar following the modes of the W3C Speech Recognition Grammar Specification [SRGS]. Defined values are "voice" and "dtmf" for DTMF input. <b>(Not Supported)</b>	
		<i>root</i>	Defines the public rule which acts as the root rule of the grammar.	
		<i>tag-format</i>	Defines the tag content format for all tags within the grammar. <b>(Not Supported)</b>	
		<i>base</i>	Declares the base URI from which relative URIs are resolved. <b>(Not Supported)</b>	
		<i>src</i>	The URI specifying the location of the grammar and optionally a rulename within that grammar, if it is external.	
		<i>scope</i>	Either "document", which makes the grammar active in all dialogs of the current document or "dialog", to make the grammar active throughout the current form.	
<b>&lt;help&gt;</b>	Catch a help event	<i>count</i>	The event count.	✓
		<i>cond</i>	An optional condition to test to see if the event is caught by this element.	
<b>&lt;if&gt;</b>	Simple conditional logic			✓
<b>&lt;link&gt;</b>	Specify a transition common to all dialogs in the link's scope	<i>next</i>	The URI to go to.	✓
		<i>expr</i>	Like next, except that the URI is dynamically determined.	
		<i>event</i>	The event to throw when the user matches one of the link grammars.	
		<i>eventexpr</i>	An ECMAScript expression evaluating to the name of the event to throw when the user matches one of the link grammars.	
		<i>message</i>	A message string providing additional context about the event being thrown.	
		<i>messageexpr</i>	An ECMAScript expression evaluating to the message string.	
		<i>dtmf</i>	The DTMF sequence for this link.	

Table K-4: VoiceXML Supported Elements &amp; Attributes (continues on pages 256 to 260)

Element	Element's Description	Parameters	Parameter's Description	Supported
<log>	Generate a debug message	<i>label</i>	A string which may be used.	✓
		<i>expr</i>	An ECMAScript expression evaluating to a string.	
<menu>	A dialog for choosing amongst alternative destinations	<i>id</i>	The identifier of the menu.	✓
		<i>scope</i>	The menu's grammar scope.	
		<i>Dtmf</i>	When set to true, the first nine choices that have not explicitly specified a value for the dtmf attribute are given the implicit ones "1", "2", etc.	
<noinput>	Catch a noinput event	<i>count</i>	The event count.	✓
		<i>cond</i>	An optional condition to test to see if the event is caught by this element.	
<nomatch>	Catch a nomatch event	<i>count</i>	The event count.	✓
		<i>cond</i>	An optional condition to test to see if the event is caught by this element.	
<object>	Interact with a custom extension	<i>name</i>	When the object is evaluated, it sets this variable to an ECMAScript value whose type is defined by the object.	✓
		<i>expr</i>	The initial value of the form item variable.	
		<i>cond</i>	An expression that must evaluate to true after conversion to Boolean in order for the form item to be visited.	
		<i>classid</i>	The URI specifying the location of the object's implementation.	
		<i>codebase</i>	The base path used to resolve relative URIs specified by classid, data, and archive. <b>(Not Supported)</b>	
		<i>codetype</i>	The content type of data expected when loading the object specified by classid. <b>(Not Supported)</b>	
		<i>data</i>	The URI specifying the location of the object's data.	
		<i>type</i>	The content type of the data specified by the data attribute. <b>(Not Supported)</b>	
<param>	Parameter in <object> or <subdialog>	<i>Name</i>	The name to be associated with this parameter when the object or sub-dialog is invoked.	✓
		<i>expr</i>	An expression that computes the value associated with name.	
		<i>value</i>	Associates a literal string value with name.	
		<i>valuetype</i>	One of data or ref, by default data; used to indicate to an object if the value associated with name is data or a URI (ref).	
		<i>type</i>	The media type of the result provided by a URI if the valuetype is ref;	
<property>	Control implementation platform settings.		For a list of available properties, refer to <a href="#">Table K-5</a> on page 260.	✓
<record>	Record an audio sample	<i>name</i>	The input item variable that holds the recording.	✓
		<i>expr</i>	The initial value of the form item variable;	
		<i>cond</i>	An expression that must evaluate to true after conversion to Boolean in order for the form item to be visited.	
		<i>modal</i>	If this is true all non-local speech and DTMF grammars are not active while making the recording.	
		<i>beep</i>	If true, a tone is emitted just prior to recording. <b>(Not Supported)</b>	

Table K-4: VoiceXML Supported Elements &amp; Attributes (continues on pages 256 to 260)

Element	Element's Description	Parameters	Parameter's Description	Supported
		<i>finalsilence</i>	The interval of silence that indicates end of speech. <b>(Not Supported)</b>	
		<i>dtmfterm</i>	If true, any DTMF keypress not matched by an active grammar is treated as a match of an active local DTMF grammar. <b>(Not Supported)</b>	
		<i>type</i>	The media format of the resulting recording.	
<b>&lt;return&gt;</b>	Return from a subdialog.	<i>event</i>	Return, and then throw this event.	✓
		<i>eventexpr</i>	Return, and then throw the event to which this ECMAScript expression evaluates.	
		<i>message</i>	A message string providing additional context about the event being thrown.	
		<i>messageexpr</i>	An ECMAScript expression evaluating to the message string.	
		<i>namelist</i>	Variable names to be returned to calling dialog.	
<b>&lt;subdialog&gt;</b>	Invoke another dialog as a subdialog of the current one	<i>name</i>	The result returned from the sub-dialog,	✓
		<i>Expr</i>	The initial value of the form item variable.	
		<i>cond</i>	An expression that must evaluate to true after conversion to Boolean in order for the form item to be visited.	
		<i>namelist</i>	The list of variables to submit. The default is to submit no variables. If a namelist is supplied, it may contain individual variable references which are submitted with the same qualification used in the namelist. Declared VoiceXML and ECMAScript variables can be referenced.	
		<i>src</i>	The URI of the sub-dialog.	
		<i>srcexpr</i>	An ECMAScript expression yielding the URI of the sub-dialog	
		<i>method</i>	See Section 5.3.8.	
<b>&lt;submit&gt;</b>	Submit values to a document server	<i>next</i>	The URI reference.	✓
		<i>expr</i>	Like next, except that the URI reference is dynamically determined.	
		<i>namelist</i>	The list of variables to submit.	
		<i>method</i>	The request method: get (the default) or post.	
<b>&lt;throw&gt;</b>	Throw an event.	<i>event</i>	The event being thrown.	✓
		<i>eventexpr</i>	An ECMAScript expression evaluating to the name of the event being thrown.	
		<i>message</i>	A message string providing additional context about the event being thrown.	
		<i>messageexpr</i>	An ECMAScript expression evaluating to the message string.	
<b>&lt;transfer&gt;</b>	Transfer the caller to another destination	<i>name</i>	Stores the outcome of a bridge transfer attempt.	✓
		<i>expr</i>	The initial value of the form item variable.	
		<i>cond</i>	An expression that must evaluate to true in order for the form item to be visited.	
		<i>dest</i>	The URI of the destination (telephone, IP telephony address).	
		<i>destexpr</i>	An ECMAScript expression yielding the URI of the destination.	
		<i>bridge</i>	Determines whether the platform remains in the connection with the caller and callee.	

**Table K-4: VoiceXML Supported Elements & Attributes (continues on pages 256 to 260)**

Element	Element's Description	Parameters	Parameter's Description	Supported
		<i>transferaudio</i>	The URI of audio source to play while the transfer attempt is in progress (before far-end answer). <b>(Not Supported)</b>	
		<i>endaudio</i>	An internal Voice Prompt ID that determines the VP that is played when the maximal time allowed for a call expires.	
		<i>finalalertaudio</i>	An internal Voice Prompt ID that determines the VP that is played <i>finalalerttime</i> seconds before the maximal time allowed for a call expires.	
		<i>finalalerttime</i>	Determines how many seconds before the maximal time allowed for a call expires, the <i>finalalertaudio</i> VP is played.	
<b>&lt;var&gt;</b>	Declare a variable	<i>name</i>	The name of the variable that holds the result.	✓
		<i>expr</i>	The initial value of the variable.	
<b>&lt;vxml&gt;</b>	Top-level element in each VoiceXML document	<i>version</i>	The version of VoiceXML of this document (required). <b>(Not Supported)</b>	✓
		<i>xmlns</i>	The designated namespace for VoiceXML (required). <b>(Not Supported)</b>	
		<i>xml:base</i>	The base URI for this document as defined in [XML-BASE]. <b>(Not Supported)</b>	
		<i>xml:lang</i>	The language identifier for this document as defined in [RFC3066]. <b>(Not Supported)</b>	
		<i>application</i>	The URI of this document's application root document, if any. <b>(Not Supported)</b>	

**Table K-5: VoiceXML Supported Properties**

Property Name	Property description	Default Value
<b>FetchTimeout</b>	The maximum time (in seconds) to wait from the time the script is fetched.	5 seconds
<b>TimeoutExpirationTime</b>	The maximum time to wait for the first digit in the user's input.	5 seconds
<b>EndDialKey</b>	The user's input that terminates the current collection.	#

## K.11 Provided Calling Card System

### K.11.1 Voice Prompts

0. "Welcome to the calling card service".
1. "To make a call press 1, for help press 2, for operator assistance press 3, to exit press 4.
2. "Invalid selection. Please try again".
3. "Unable to recognize your entry. Please try again" .
4. "Please call again later".
5. "Please enter your account number followed by the pound key".
6. "Please enter your password followed by the pound key".
7. "Please wait while we verify your account number and password".

8. "Your account number and password do not match".
9. "We are having technical difficulties, please call again later".
10. "Please enter the number that you wish to call followed by the pound key".
11. "The number you are calling is busy".
12. "The number you are calling is not answering, please call again later".
13. "The person you called has hung up".
14. "We are unable to complete your call".
15. "Transferring your call".
16. "Thank you for using the calling card service".
17. "The menu is currently empty".
18. "Operator assistance is currently unavailable".
19. "You are unauthorized to access the number you are attempting to call".
20. Final alert (one beep at 640 Hz for 0.5 second duration)
21. Time of call expired (one beep at 640 Hz for 1.5 second duration)

### K.11.2 VXML Flow Chart

Figure K-7: VXML Script Opening Menu

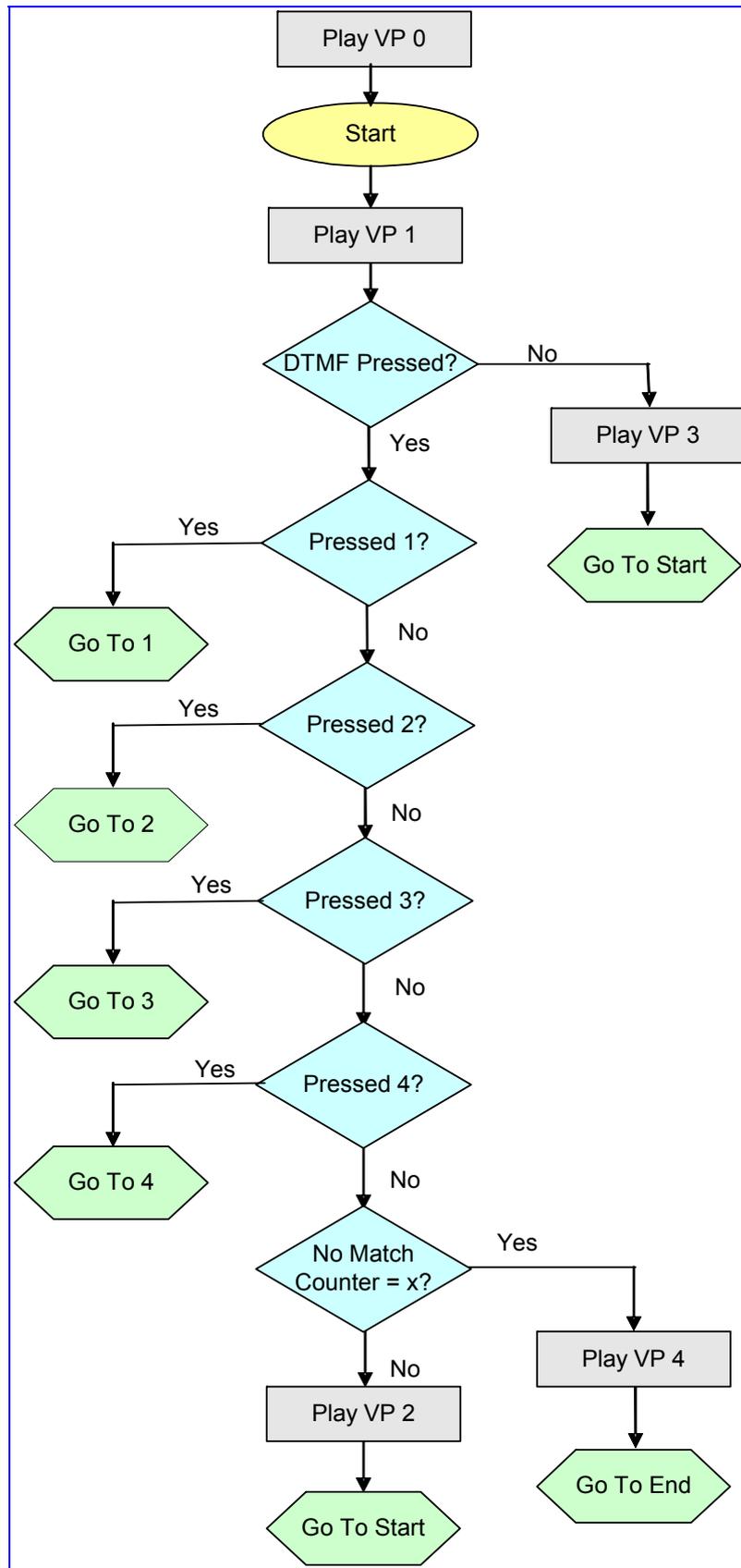


Figure K-8: VXML Script Option 1, Make a Call

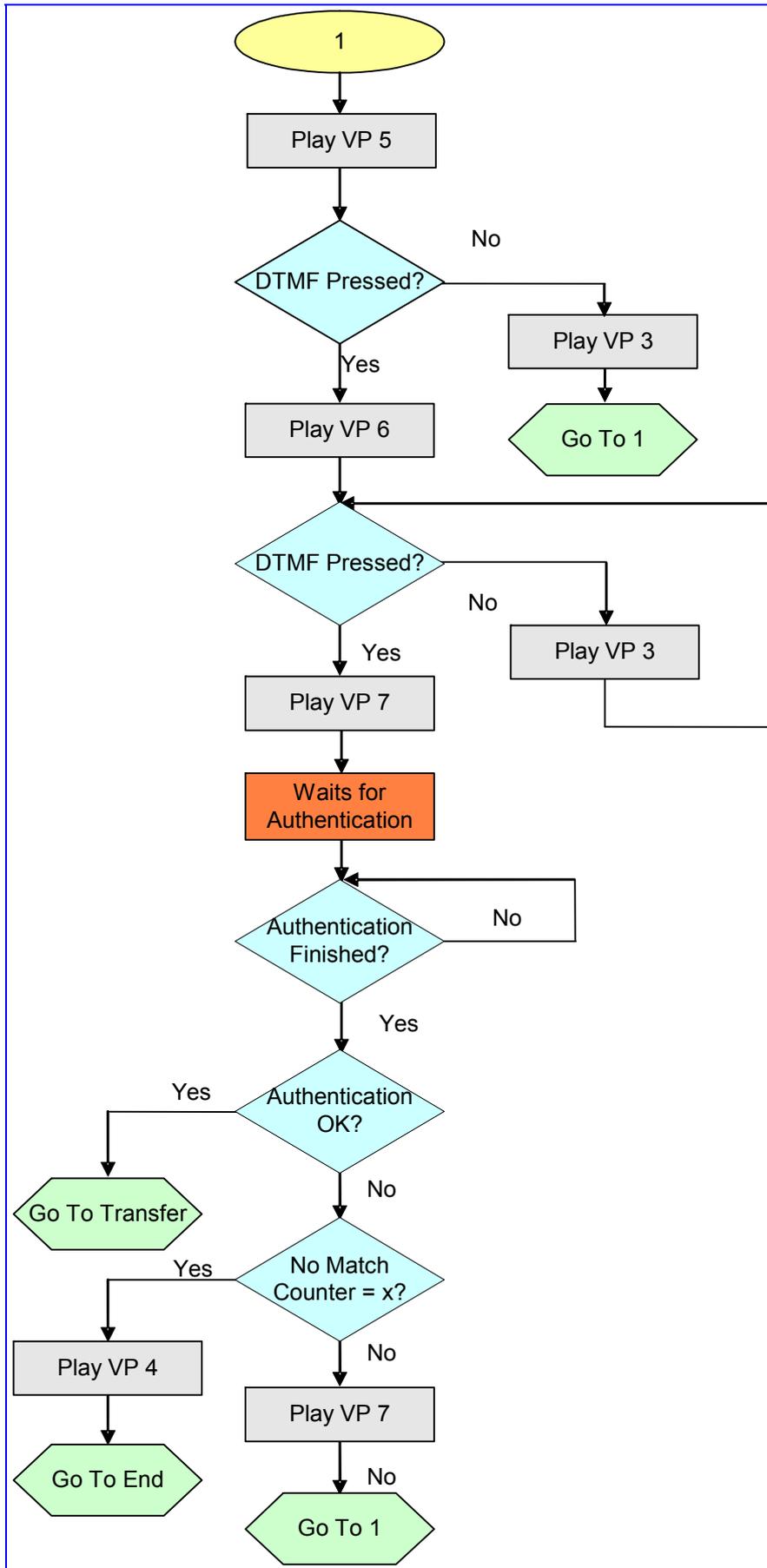


Figure K-9: VXML Script, Call Transfer Procedure

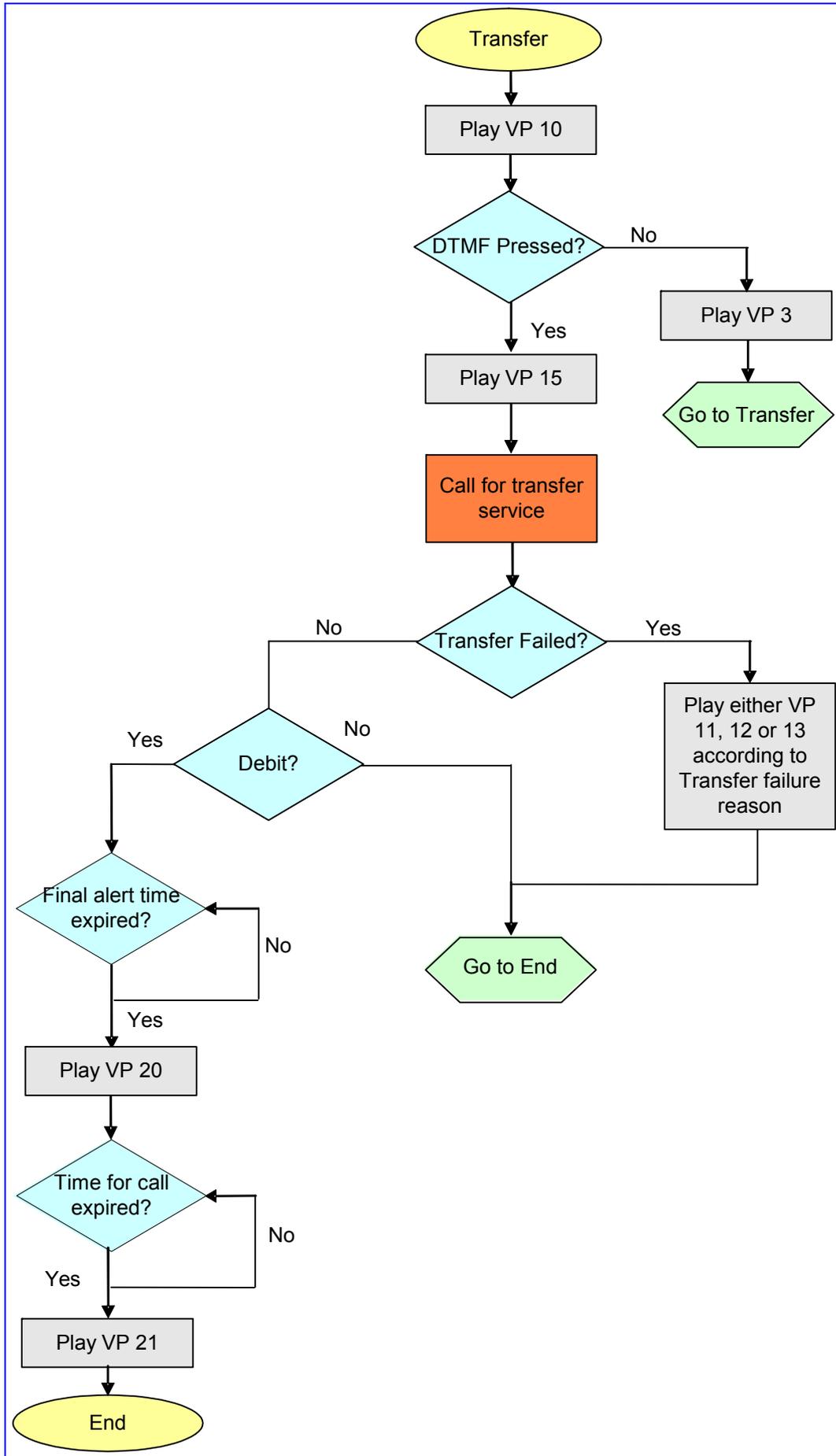


Figure K-10: VXML Script, Options 2, 3 and 4

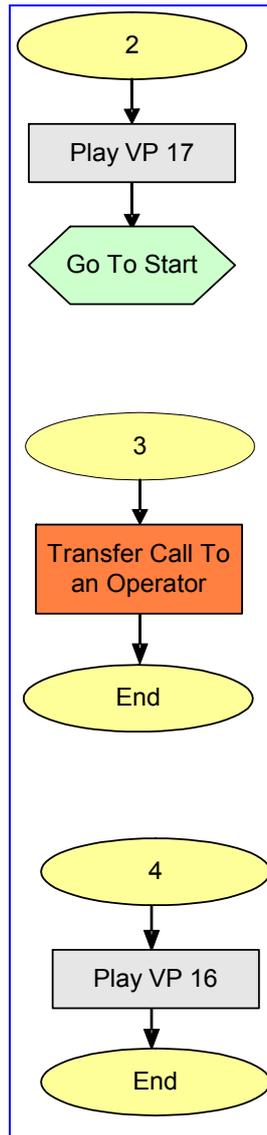
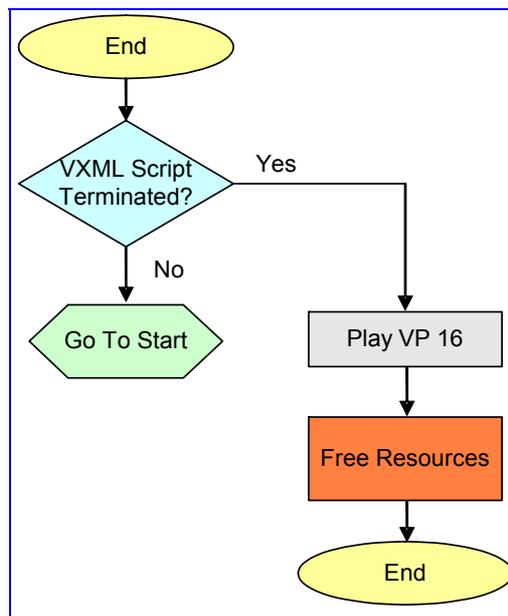


Figure K-11: VXML Script, Call Termination



## K.12 VXML Script Example

**Figure K-12: VXML Script Example (continues on pages 266 to 270)**

```

<?xml version="1.0" encoding="UTF-8"?>
<vxml version="1.0" application="http://phoenix1.iperia.com:8080/sa3/jsp/sa.jsp">
<var name="AAStatus" expr="0"/>

  <form id="main_mc">
    <log label="starting main form"/>
    <block name="x">
      <prompt>
        <audio src="/0.wav">
          wellcome to the pre-paid call service
        </audio>
      </prompt>

      <goto next="#main_menu"/>
    </block>

  </form>

  <form id="main_menu">
    <log label="starting main_menu form"/>
    <field name="option" type="number">
      <dtmf>
        1 | 2 | 3 | 4
      </dtmf>
      <property name="TimeoutExpirationTime" value="1"/>

      <prompt bargein="true">
        <audio src="/1.wav">
          for making a call press 1
          for help press 2
          for human service press 3
          to exit press 4
        </audio>
      </prompt>

      <nomatch>
        <log label="try again interupt"/>

        <prompt>
          <audio src="/2.wav">
            that is an invalid selection. Please try again
          </audio>
        </prompt>
      </nomatch>

      <noinput>
        <log label="no try interupt"/>

        <prompt>
          <audio src="/3.wav">
            we did not get your input. Please try again
          </audio>
        </prompt>
      </noinput>

      <catch event="noinput nomatch" count="4">
        <log label="quit interupt"/>

        <prompt>
          <audio src="/4.wav">
            sorry please try again later
          </audio>
        </prompt>
        <log label="please try again later"/>

        <goto next="#disconnect"/>
      </catch>
    </field>
    <filled>
      <if cond="option == '1'">
    
```

**Figure K-12: VXML Script Example (continues on pages 266 to 270)**

```

        <if cond="AAStatus==0">
            <goto next="#PerformAuthen"/>
        </if>
        <else/>
            <goto next="#PerformTransfer"/>
        </else/>

        </if>

        <elseif cond="option =='2'"/>
            <prompt>
                <audio src="/17.wav">
                    sorry help is currently unavailble
                </audio>
            </prompt>
            <goto next="#main_menu"/>
        </elseif>

        <elseif cond="option =='3'"/>
            <goto next="#HelpTransfer"/>
        </elseif>

        <elseif cond="option == '4'"/>
            <goto next="#disconnect"/>
        </elseif>
    </filled>
</field>
</form>

<form id="GetUserInfo">
    <log label="starting get user info form" cond="'1' == '1'"/>
    <field name="user_account_num" type="digits">
        <prompt bargein="true">
            <audio src="/5.wav">
                please enter your account number
            </audio>
        </prompt>
    </field>

    <field name="user_passwd" type="digits">
        <prompt bargein="true">
            <audio src="/6.wav">
                please enter your pin number
            </audio>
        </prompt>
    </field>

    <noinput>
        <log label="no try interupt"/>

        <prompt>
            <audio src="/3.wav">
                we did not get your input. Please try again
            </audio>
        </prompt>
    </noinput>

    <catch event="noinput nomatch" count="3">
        <prompt>
            <audio src="/4.wav">
                please try again later
            </audio>
        </prompt>

        <goto next="#disconnect"/>
    </catch>

    <filled>
        <return namelist="user_account_num user_passwd"/>
    </filled>
</form>

```

**Figure K-12: VXML Script Example (continues on pages 266 to 270)**

```

</form>

<form id="GetCalledPartyTelephone">
  <log label="starting get called party telephone form" cond="'1' == '1'"/>
  <field name="dest_number" type="digits">
    <prompt bargein="true">
      <audio src="/10.wav">
        please dial the telephone number

      </audio>
    </prompt>

    <noinput>

      <prompt>

        <audio src="/3.wav">
          illegal input

        </audio>
      </prompt>

    </noinput>

    <catch event="noinput nomatch" count="3">
      <goto next="#disconnect"/>
    </catch>

    <filled>
      <return namelist="dest_number"/>

    </filled>

  </field>

</form>

<form id="HelpTransfer">
  <log label="performing help transfer"/>

  <transfer name="mycall" destexpr="201" bridge="true">

    <filled>
      <goto next="#disconnect"/>

    </filled>

  </transfer>

</form>

<form id="PerformAuthen">

  <log label="performing authentication"/>
  <subdialog name="AuthenticationInfo" src="#GetUserInfo">

    <filled>
      <prompt>

        <audio src="/7.wav">
          please wait while your account and pin numbers are
          checked

        </audio>
      </prompt>

    </filled>

  </subdialog>

  <object name="authenticate" classid="builtin://com.audiocodes.aaa.authenticate">
    <param name="account" expr="AuthenticationInfo.user_account_num"/>

```

**Figure K-12: VXML Script Example (continues on pages 266 to 270)**

```

<param name="password" expr="AuthenticationInfo.user_passwd"/>

<nomatch>

    <prompt>
        <audio src="/8.wav">
            your account and password did not match.
        </audio>
    </prompt>
    <clear namelist="AuthenticationInfo authenticate"/>

</nomatch>

<catch event="nomatch" count="4">
    <goto next="#disconnect"/>
</catch>

<filled>

    <assign name="AAStatus" expr="1"/>
    <goto next="#PerformTransfer"/>

</filled>
</object>

</form>

<form id="PerformTransfer">
    <log label="starting transfer form"/>
    <subdialog name="Call" src="#GetCalledPartyTelephone">
</subdialog>

    <object name="authorize" classid="builtin://com.audiocodes.aaa.authorize">
        <param name="dest" expr="Call.dest_number"/>

        <nomatch>

            <prompt>
                <audio src="/9.wav"/>
                    you are not authorized.
            </prompt>
            <clear namelist=" Call authorize "/>

        </nomatch>

        <catch event="nomatch" count="3">
            <goto next="#disconnect"/>
        </catch>

        <filled>
            <prompt>
                <audio src="/15.wav">
                    calling
                </audio>
            </prompt>

        </filled>
    </object>

    <transfer name="mycall" destexpr="Call.dest_number" bridge="true" endaudio="7"
finalalertaudio="4" finalalerttime="10">

        <filled>
            <if cond="mycall.Result=='fail'">

                <prompt>
                    <audio src="/14.wav">

```

**Figure K-12: VXML Script Example (continues on pages 266 to 270)**

```

        called transfer failed
    </audio>
</prompt>
<goto next="#main_menu"/>

<elseif cond="mycall.Result=='busy'"/>
    <prompt>
        <audio src="/11.wav">
            called party is busy
        </audio>
    </prompt>
    <goto next="#main_menu"/>

<elseif cond="mycall.Result=='maxtime'"/>
    <prompt>
        <audio src="/7.wav">
            reached maximum time allowed
        </audio>
    </prompt>
    <goto next="#main_menu"/>

    <else/>

    <goto next="#main_menu"/>

</if>

</filled>

</transfer>

</form>

<form id="disconnect">
    <block name="x">
        <prompt>
            <audio src="/16.wav">
                goodbye
            </audio>
        </prompt>
        <log label="disconnect"/>
        <exit/>
    </block>
</form>

</vxml>

```

## Appendix L SNMP Traps

This section provides information on proprietary SNMP traps currently supported by the gateway. There is a separation between traps that are alarms and traps that are not (logs). Currently all have the same structure made up of the same 10 varbinds (Variable Binding) (1.3.6.1.4.1.5003.9.10.1.21.1).

### L.1 Alarm Traps

The following tables provide information on alarms that are raised as a result of a generated SNMP trap. The component name (described in each of the following headings) refers to the string that is provided in the 'acBoardTrapGlobalsSource' trap varbind. To clear a generated alarm the same notification type is sent but with the severity set to 'cleared'.

#### L.1.1 Component: System#0

**Table L-1: acBoardFatalError Alarm Trap**

<b>Alarm:</b>	acBoardFatalError
<b>OID:</b>	1.3.6.1.4.1.5003.9.10.1.21.2.0.1
<b>Default Severity</b>	Critical
<b>Event Type:</b>	equipmentAlarm
<b>Probable Cause:</b>	underlyingResourceUnavailable (56)
<b>Alarm Text:</b>	Board Fatal Error: <text>
<b>Status Changes:</b>	
Condition:	Any fatal error
Alarm status:	Critical
<text> value:	A run-time specific string describing the fatal error
Condition:	After fatal error
Alarm status:	Status stays critical until reboot. A clear trap is not sent.
Corrective Action:	Capture the alarm information and the Syslog clause, if active. Contact your first-level support group. The support group will likely want to collect additional data from the device and perform a reset.

**Table L-2: acBoardConfigurationError Alarm Trap**

<b>Alarm:</b>	acBoardConfigurationError
<b>OID:</b>	1.3.6.1.4.1.5003.9.10.1.21.2.0.2
<b>Default Severity</b>	Critical
<b>Event Type:</b>	equipmentAlarm
<b>Probable Cause:</b>	underlyingResourceUnavailable (56)
<b>Alarm Text:</b>	Board Config Error: <text>
<b>Status Changes:</b>	
Condition:	A configuration error was detected
Alarm status:	critical
<text> value:	A run-time specific string describing the configuration error.
Condition:	After configuration error
Alarm status:	Status stays critical until reboot. A clear trap is not sent.

**Table L-2: acBoardConfigurationError Alarm Trap**

Corrective Action:	Inspect the run-time specific string to determine the nature of the configuration error. Fix the configuration error using the appropriate tool: Web interface, EMS, or <i>ini</i> file. Save the configuration and if necessary reset the device.
--------------------	--

**Table L-3: acBoardTemperatureAlarm Alarm Trap**

<b>Alarm:</b>	acBoardTemperatureAlarm
<b>OID:</b>	1.3.6.1.4.1.5003.9.10.1.21.2.0.3
<b>Default Severity</b>	Critical
<b>Event Type:</b>	equipmentAlarm
<b>Probable Cause:</b>	temperatureUnacceptable (50)
<b>Alarm Text:</b>	Board temperature too high
<b>Status Changes:</b>	
Condition:	Temperature is above 60 degrees C (140 degrees F)
Alarm status:	Critical
Condition:	After raise, temperature falls below 55 degrees C (131 degrees F)
Alarm status:	Cleared
Corrective Action:	Inspect the system. Determine if all fans in the system are properly operating.

**Table L-4: acBoardEvResettingBoard Alarm Trap**

<b>Alarm:</b>	acBoardEvResettingBoard
<b>OID:</b>	1.3.6.1.4.1.5003.9.10.1.21.2.0.5
<b>Default Severity</b>	Critical
<b>Event Type:</b>	equipmentAlarm
<b>Probable Cause:</b>	outOfService (71)
<b>Alarm Text:</b>	User resetting board
<b>Status Changes:</b>	
Condition:	When a soft reset is triggered via the Web interface or SNMP.
Alarm status:	Critical
Condition:	After raise
Alarm status:	Status stays critical until reboot. A clear trap is not sent.
Corrective Action:	A network administrator has taken action to reset the device. No corrective action is required.

**Table L-5: acFeatureKeyError Alarm Trap**

<b>Alarm:</b>	acFeatureKeyError
<b>OID:</b>	1.3.6.1.4.1.5003.9.10.1.21.2.0.6
<b>Default Severity</b>	Critical
<b>Event Type:</b>	processingErrorAlarm
<b>Probable Cause:</b>	configurationOrCustomizationError (7)
<b>Alarm Text:</b>	Feature key error
<b>Status Changes:</b>	
Note:	Support of this alarm is pending
Condition:	A feature key error has been detected
Alarm status:	Critical

**Table L-5: acFeatureKeyError Alarm Trap**

Condition:	After raise
Alarm status:	Status stays critical until reboot. A clear trap is not sent.
Corrective Action:	Obtain a corrected feature key from AudioCodes, activate it by loading to the device, then reset the device.

**Table L-6: acBoardCallResourcesAlarm Alarm Trap**

<b>Alarm:</b>	acBoardCallResourcesAlarm
<b>OID:</b>	1.3.6.1.4.1.5003.9.10.1.21.2.0.8
<b>Default Severity</b>	Major
<b>Event Type:</b>	processingErrorAlarm
<b>Probable Cause:</b>	softwareError (46)
<b>Alarm Text:</b>	Call resources alarm
<b>Status Changes:</b>	
Condition:	Number of free channels exceeds the predefined RAI <i>high</i> threshold.
Alarm Status:	Major
Note:	To enable this alarm the RAI mechanism must be activated (EnableRAI = 1).
Condition:	Number of free channels falls below the predefined RAI <i>low</i> threshold.
Alarm Status:	Cleared

**Table L-7: acBoardControllerFailureAlarm Alarm Trap**

<b>Alarm:</b>	acBoardControllerFailureAlarm
<b>OID:</b>	1.3.6.1.4.1.5003.9.10.1.21.2.0.9
<b>Default Severity</b>	Minor
<b>Event Type:</b>	processingErrorAlarm
<b>Probable Cause:</b>	softwareError (46)
<b>Alarm Text:</b>	Controller failure alarm
<b>Status Changes:</b>	
Condition:	Proxy has not been found
Alarm Status:	Major
Additional Info:	Proxy not found. Use internal routing or Proxy lost. looking for another Proxy
Condition:	Proxy is found. The clear message includes the IP address of this Proxy.
Alarm Status:	Cleared

**Table L-8: acBoardOverloadAlarm Alarm Trap**

<b>Alarm:</b>	acBoardOverloadAlarm
<b>OID:</b>	1.3.6.1.4.1.5003.9.10.1.21.2.0.11
<b>Default Severity</b>	Major
<b>Event Type:</b>	processingErrorAlarm
<b>Probable Cause:</b>	softwareError (46)
<b>Alarm Text:</b>	Board overload alarm
<b>Status Changes:</b>	

**Table L-8: acBoardOverloadAlarm Alarm Trap**

Condition:	An overload condition exists in one or more of the system components.
Alarm Status:	Major
Condition:	The overload condition passed
Alarm Status:	Cleared

## L.1.2 Component: AlarmManager#0

**Table L-9: acActiveAlarmTableOverflow Alarm Trap**

<b>Alarm:</b>	acActiveAlarmTableOverflow
<b>OID:</b>	1.3.6.1.4.15003.9.10.1.21.2.0.12
<b>Default Severity</b>	Major
<b>Event Type:</b>	processingErrorAlarm
<b>Probable Cause:</b>	resourceAtOrNearingCapacity (43)
<b>Alarm Text:</b>	Active alarm table overflow
<b>Status Changes:</b>	
Condition:	Too many alarms to fit in the active alarm table
Alarm status:	major
Condition:	After raise
Alarm status:	Status stays major until reboot. A clear trap is not sent.
Note:	The status stays major until reboot as it denotes a possible loss of information until the next reboot. If an alarm is raised when the table is full, it is possible that the alarm is active, but does not appear in the active alarm table.
Corrective Action:	Some alarm information may have been lost, but the ability of the device to perform its basic operations has not been impacted. A reboot is the only way to completely clear a problem with the active alarm table. Contact your first-level group.

## L.1.3 Component: EthernetLink#0

**Table L-10: acBoardEthernetLinkAlarm Alarm Trap**

<b>Alarm:</b>	acBoardEthernetLinkAlarm
<b>OID:</b>	1.3.6.1.4.1.5003.9.10.1.21.2.0.10
<b>Default Severity</b>	Critical
<b>Event Type:</b>	equipmentAlarm
<b>Probable Cause:</b>	underlyingResourceUnavailable (56)
<b>Alarm Text:</b>	Ethernet link alarm: <text>
<b>Status Changes:</b>	
Condition:	Fault on single interface
Alarm status:	major
<text> value:	Redundant link is down
Condition:	Fault on both interfaces
Alarm status:	critical
<text> value:	No Ethernet link
Condition:	Both interfaces are operational
Alarm status:	cleared
Corrective Action:	Ensure that both Ethernet cables are plugged into the back of the system. Inspect the system's Ethernet link lights to determine which interface is failing. Reconnect the cable or fix the network problem

## L.1.4 Other Traps

The following are provided as SNMP traps and are not alarms.

**Table L-11: coldStart Trap**

<b>Trap Name:</b>	coldStart
<b>OID:</b>	1.3.6.1.6.3.1.1.5.1
<b>MIB</b>	SNMPv2-MIB
<b>Note:</b>	This is a trap from the standard SNMP MIB.

**Table L-12: authenticationFailure Trap**

<b>Trap Name:</b>	authenticationFailure
<b>OID:</b>	1.3.6.1.6.3.1.1.5.5
<b>MIB</b>	SNMPv2-MIB

**Table L-13: acBoardEvBoardStarted Trap**

<b>Trap Name:</b>	acBoardEvBoardStarted
<b>OID:</b>	1.3.6.1.4.1.5003.9.10.1.21.2.0.4
<b>MIB</b>	AcBoard
<b>Severity</b>	cleared
<b>Event Type:</b>	equipmentAlarm
<b>Probable Cause:</b>	Other(0)
<b>Alarm Text:</b>	Initialization Ended
<b>Note:</b>	This is the AudioCodes Enterprise application cold start trap.

## L.1.5 Trap Varbinds

Each trap described above provides the following fields (known as 'varbinds'). Refer to the AcBoard MIB for additional details on these varbinds.

- acBoardTrapGlobalsName
- acBoardTrapGlobalsTextualDescription
- acBoardTrapGlobalsSource
- acBoardTrapGlobalsSeverity
- acBoardTrapGlobalsUniqID
- acBoardTrapGlobalsType
- acBoardTrapGlobalsProbableCause
- acBoardTrapGlobalsAdditionalInfo1
- acBoardTrapGlobalsAdditionalInfo2
- acBoardTrapGlobalsAdditionalInfo3

Note that 'acBoardTrapGlobalsName' is actually a number. The value of this varbind is 'X' minus 1, where 'X' is the last number in the trap's OID. For example, the 'name' of 'acBoardEthernetLinkAlarm' is '9'. The OID for 'acBoardEthernetLinkAlarm' is 1.3.6.1.4.1.5003.9.10.1.21.2.0.10.

## Appendix M Regulatory Information

<i>Declaration of Conformity</i>	
<b>Application of Council Directives:</b>	73/23/EEC (including amendments), 89/336/EEC (including amendments), 1999/5/EC Annex-II of the Directive
<b>Standards to which Conformity is Declared:</b>	EN55022: 1998, Class A EN55024: 1998 EN61000-3-2: 1995 (including amendments A1: 1998, A2: 1998, A14: 2000) EN61000-3-3: 1995 EN60950-1: 2001 TBR-4: 1995 (including amendment A1: 1997) TBR-13: 1996 TBR-12: 1993 (including amendment 1: 1996)
<b>Manufacturer's Name:</b>	AudioCodes Ltd.
<b>Manufacturer's Address:</b>	1 Hayarden Street, Airport City, Lod 70151, Israel.
<b>Type of Equipment:</b>	Digital VoIP System.
<b>Model Numbers:</b>	<b>Mediant 2000, Stretto 2000, IPmedia 2000</b>
I, the undersigned, hereby declare that the equipment specified above conforms to the above Directives and Standards.	
 _____ <i>Signature</i>	11 <sup>th</sup> February, 2005 _____ <i>Date (Day/Month/Year)</i>
	Airport City, Lod, Israel _____ <i>Location</i>
I. Zusmanovich, Compliance Engineering Manager	

Czech	[AudioCodes Ltd] tímto prohlašuje, že tento [2000 Series] je ve shodě se základními požadavky a dalšími příslušnými ustanoveními směrnice 1999/5/ES."
Danish	Undertegnede [AudioCodes Ltd] erklærer herved, at følgende udstyr [2000 Series] overholder de væsentlige krav og øvrige relevante krav i direktiv 1999/5/EF
Dutch	Hierbij verklaart [AudioCodes Ltd] dat het toestel [2000 Series] in overeenstemming is met de essentiële eisen en de andere relevante bepalingen van richtlijn 1999/5/EG
English	Hereby, [AudioCodes Ltd], declares that this [2000 Series] is in compliance with the essential requirements and other relevant provisions of Directive 1999/5/EC.
Estonian	Käesolevaga kinnitab [AudioCodes Ltd] seadme [2000 Series] vastavust direktiivi 1999/5/EÜ põhinõuetele ja nimetatud direktiivist tulenevatele teistele asjakohastele sätetele.
Finnish	[AudioCodes Ltd] vakuuttaa täten että [2000 Series] tyypinen laite on direktiivin 1999/5/EY oleellisten vaatimusten ja sitä koskevien direktiivin muiden ehtojen mukainen.
French	Par la présente [AudioCodes Ltd] déclare que l'appareil [2000 Series] est conforme aux exigences essentielles et aux autres dispositions pertinentes de la directive 1999/5/CE
German	Hiermit erkläre [AudioCodes Ltd], dass sich dieser/diese/dieses [2000 Series] in Übereinstimmung mit den grundlegenden Anforderungen und den anderen relevanten Vorschriften der Richtlinie 1999/5/EG befindet". (BMW)
Greek	ΜΕ ΤΗΝ ΠΑΡΟΥΣΑ [AudioCodes Ltd] ΔΗΛΩΝΕΙ ΟΤΙ [2000 Series] ΣΥΜΜΟΡΦΩΝΕΤΑΙ ΠΡΟΣ ΤΙΣ ΟΥΣΙΩΔΕΙΣ ΑΠΑΙΤΗΣΕΙΣ ΚΑΙ ΤΙΣ ΛΟΙΠΕΣ ΣΧΕΤΙΚΕΣ ΔΙΑΤΑΞΕΙΣ ΤΗΣ ΟΔΗΓΙΑΣ 1999/5/ΕΚ
Hungarian	Alulírott, [AudioCodes Ltd] nyilatkozom, hogy a [2000 Series] megfelel a vonatkozó alapvető követelményeknek és az 1999/5/EC irányelv egyéb előírásainak
Icelandic	æki þetta er í samræmi við tilskipun Evrópusambandsins 1999/5
Italian	Con la presente [AudioCodes Ltd] dichiara che questo (2000 Series) è conforme ai requisiti essenziali ed alle altre disposizioni pertinenti stabilite dalla direttiva 1999/5/CE.
Latvian	Ar šo [AudioCodes Ltd] deklarē, ka [2000 Series] atbilst Direktīvas 1999/5/EK būtiskajām prasībām un citiem ar to saistītajiem noteikumiem.
Lithuanian	[AudioCodes Ltd] deklaruoja, kad irenginys [2000 Series] tenkina 1999/5/EB Direktyvos esminius reikalavimus ir kitas šios direktyvos nuostatas
Maltese	Hawnhekk, [AudioCodes Ltd], jiddikjara li dan [2000 Series] jikkonforma mal-htigijiet essenzjali u ma provvedimenti oħrajn relevanti li hemm fid-Direttiva 1999/5/EC
Norwegian	Dette produktet er i samhörighet med det Europeiske Direktiv 1999/5
Polish	[AudioCodes Ltd], deklaruje, że pełna odpowiedzialność, że wyrób [2000 Series] spełnia podstawowe wymagania i odpowiada warunkom zawartym w dyrektywie 1999/5/EC
Portuguese	[AudioCodes Ltd] declara que este [2000 Series] está conforme com os requisitos essenciais e outras disposições da Directiva 1999/5/CE.
Slovak	[AudioCodes Ltd] týmto vyhlasuje, že [2000 Series] spĺňa základné požiadavky a všetky príslušné ustanovenia Smernice 1999/5/ES.
Slovene	Šiuo [AudioCodes Ltd] deklaruoja, kad šis [2000 Series] atitinka esminius reikalavimus ir kitas 1999/5/EB Direktyvos nuostatas.
Spanish	Por medio de la presente [AudioCodes Ltd] declara que el (200 Series) cumple con los requisitos esenciales y cualesquiera otras disposiciones aplicables o exigibles de la Directiva 1999/5/CE
Swedish	Härmed intygar [AudioCodes Ltd] att denna [2000 Series] står i överensstämmelse med de väsentliga egenskapskrav och övriga relevanta bestämmelser som framgår av direktiv 1999/5/EG.

### Safety Notices

Installation and service of this gateway must only be performed by authorized, qualified service personnel.  
The protective earth terminal on the back of the 2000 must be permanently connected to protective earth.

## Industry Canada Notice

This equipment meets the applicable Industry Canada Terminal Equipment technical specifications. This is confirmed by the registration numbers. The abbreviation, IC, before the registration number signifies that registration was performed based on a declaration of conformity indicating that Industry Canada technical specifications were met. It does not imply that Industry Canada approved the equipment.

## Digital Device Warnings

This equipment complies with Part 68 of the FCC rules and the requirements adopted by ACTA. On the bottom of this equipment is a label that contains a product identifier in the format US:AC1ISNANMED2KDC. If requested this number must be provided to the telephone company.

The Telephone company may make changes in the facilities, equipment, operations or procedures that could affect the operation of the equipment. If this happens the telephone company will provide advance notice in order for you to make necessary modifications to maintain uninterrupted service. Should you experience trouble with this telephone equipment, contact: *AudioCodes Inc, San Jose, CA USA. Tel: 1 408 577 0488. Do not attempt to repair this equipment!*

**Do not attempt to repair this equipment!**

**Facility Interface Code:**            **04DU9.BN, 04DU9.DN, 04DU9.1KN, 4DU9.ISN**  
**Service Order Code:**            **6.0N**  
**USOC Jack Type:**                 **RJ21X or RJ48C**

If this gateway causes harm to the telephone network, the telephone company will notify you in advance that temporary discontinuance of service may be required. But if advance notice isn't practical, the telephone company will notify the customer as soon as possible. Also you will be advised of your right to file complaint with the FCC if you believe it is necessary.

## Network Information and Intent of Use

The products are for access to ISDN at 2048 kb/s and for access to G.703 Leased lines at 2048 kb/s.

### Network Compatibility

The products support the Telecom networks in EU that comply with TBR4 and TBR13.

### Telecommunication Safety

The safety status of each port is declared and detailed in the table below:

Ports	Safety Status
E1 or T1	TNV-1
Ethernet (100 Base-TX)	SELV

**TNV-1:**    Telecommunication network voltage circuits whose normal operating voltages do not exceed the limits for SELV under normal operating conditions and on which over voltages from telecommunication networks are possible.

**SELV:**     Safety extra low voltage circuit.

## FCC Statement

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.



## **AudioCodes Offices**

### **International Headquarters**

AudioCodes Ltd.

1 Hayarden Street, Airport City, Lod 70151, Israel.

Tel: +972-3-976 4000

Fax: +972-3-976 4040

Email: [info@audiocodes.com](mailto:info@audiocodes.com)

### **USA Headquarters**

AudioCodes, Inc.

2099 Gateway Place, Suite 500

San Jose, CA 95110

Tel: +1-408-411-1175

Fax: +1-408-451-9520

Email: [info@audiocodes.com](mailto:info@audiocodes.com)

### **USA Offices**

Boston (MA), Chicago (IL), Research Triangle Park (NC),

Richardson (TX), Somerset (NJ)

### **AudioCodes Offices Worldwide**

Beijing, London, Mexico City, Paris, Tokyo

[info@audiocodes.com](mailto:info@audiocodes.com)

[www.audiocodes.com](http://www.audiocodes.com)

