



Reference Manual

Mediatrix[®] 4100 Series

Mediatrix 4108 / 4116 / 4124 (FXS)

SIP Version

Product Version 5.0

Document Revision 21

July 26, 2012



Media5 Corporation
4229 Garlock Street
Sherbrooke, Québec, Canada J1L 2C8

Mediatrix® 4100 Series Reference Manual

© 2012, Media5 Corporation

All rights reserved. No part of this publication may be reproduced or used in any form or by any means – graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems – without the express written permission of the publisher.

Media5 Corporation reserves the right to revise this publication and make changes at any time and without the obligation to notify any person and/or entity of such revisions and/or changes.

Trademarks

Microsoft and Windows are registered trademarks of Microsoft Corporation.

Adobe and Acrobat are registered trademarks of Adobe Systems Incorporated.

All other trademarks and registered trademarks are the property of their respective owners.

Third-Party Software Copyright Information

The Mediatrix® 4100 Series firmware aggregates some third-party software modules (open source and commercial) that are distributed to you in accordance with their respective licenses. Refer to the *Third Party Software Copyright Information* addendum available on the Mediatrix Download Portal, which lists the third-party software modules along with any copyright and license information.

Contents

Preface

About this Manual	xvii
Document Objectives.....	xvii
Intended Audience.....	xvii
Related Documentation	xviii
Document Structure.....	xviii
Document Conventions	xx
Warning Definition.....	xx
Where to find Translated Warning Definition.....	xx
Other Conventions	xxi
SCN vs PSTN.....	xxi
Standards Supported	xxi
Obtaining Documentation	xxi
Media5 Web Site	xxi
Media5 Download Portal	xxi
Documentation Feedback	xxii
Unit Manager Network – Element Management System.....	xxii
End User Technical Support.....	xxii

Installation and Web Page Configuration

Chapter 1

Installation	3
Requirements	3
Safety Recommendations.....	3
Package Contents	3
Overview.....	4
About the Mediatix 4100	4
Placing a Call	4
Management Choices	6
Panels.....	7
Front Indicators	7
Rear Connectors	9
Choosing a Suitable Installation Site	9
Location.....	9
Rack-Mounting	10
Condensation	10
Cleaning	10
Hardware Connection	10
Reserving an IP Address.....	10
Connecting Cables – Mediatix 4108/4116 Models.....	11
Connecting Cables – Mediatix 4124 Model.....	12
Starting the Mediatix 4100 for the First Time.....	13
IP Address Discovery or Configuration	13
Initial Provisioning Sequence	14

Special Vocal Features	15
LED Behaviour in Starting Mode	15
LED Indicators	15
Ready LED	15
In Use LED	15
LAN LED	15
Power LED	16
LED Patterns	16
Bootng LED Pattern Description.....	17
NormalMode LED Pattern Description	17
AdminMode LED Pattern Description.....	18
Recovery Mode LED Patterns.....	18
Reset / Default Switch	19
At Run-Time	19
At Start-Time	19
Recovery Mode	20
Factory Reset.....	21
Software Restart.....	22
Restart Behaviour.....	22
Verifying the Installation.....	23

Chapter 2

Web Interface – Introduction.....	25
Introduction	25
Using the Web Interface	25
Web Interface Access Limitation	25
System Status of the Mediatrix 4100.....	27
Menu Frame	27
Content Frame	28
Submitting Changes	28
Syslog Monitoring	29
Configuring the Syslog Daemon Application	30

Chapter 3

Web Interface – Management.....	31
Admin Page	31
HTTP Server Password.....	31
System Management	33
Group Port Management.....	34
Interface Management	34
Network Settings.....	35
Ethernet Connection Speed	35
Network Settings	36
SNTP Settings.....	37
Configuration File Download.....	40
Configuration File Download Server.....	40
Configuration File Server Settings.....	41
Setting up the Configuration File Download	42
Configuration Files Encryption	45
Configuration Download Procedure	46
Automatic Configuration Update	46
Error Handling	49
Firmware Download.....	51
Before Downloading	51

Firmware Servers Configuration.....	52
Setting up the Firmware Download	53
Firmware Download Procedure.....	58
Automatic Firmware Update.....	58
Spanning Tree Protocol (STP)	61
Firmware Downgrade	61
Emergency Firmware Procedure.....	61

Chapter 4

Web Interface – SIP Parameters 63

SIP Servers Configuration	63
SIP Servers	63
SIP Configuration	64
SIP User Agent.....	65
SIP Registration	66
SIP Publication	67
SIP Interop.....	67
SIP Penalty Box	67
SIP Transport Type	68
Interop Parameters.....	69
SIP Authentication	71

Chapter 5

Web Interface – Telephony..... 73

Digit Maps.....	73
Syntax	73
Special Characters	74
How to Use a Digit Map.....	74
General Parameters	76
Allowed Digit Maps	77
Blocked Digit Maps.....	78
Voice & Fax Codecs	79
G.711 PCMA and PCMU.....	79
G.726.....	80
G.723.1.....	80
G.729.....	80
General Parameters	81
G.711 Codec Parameters.....	84
G.729 Codec Parameters.....	85
G.723 Codec Parameters.....	85
G.726 Codecs Parameters.....	86
Fax Parameters	87
Call Forward	90
On Busy.....	91
On No Answer	93
Unconditional.....	94
Services	96
Call Transfer.....	96
Call Waiting	97
Conference.....	100
Call Hold	102
Second Call	103
Automatic Call	103
Miscellaneous	104

Country Selection	104
Custom Tone Configuration	105
Message Waiting Indicator	112

Chapter 6

Web Interface – Advanced..... 117

Quality of Service (QoS)	117
802.1q Configuration	117
DiffServ Configuration	119
Emergency Page	120
Emergency Call Configuration.....	120
STUN Configuration.....	121
SIP Outbound Proxy.....	121
Restrictions on the Media5 STUN Implementation	121
STUN Client Configuration	122
SIP Custom NAT Traversal	123

SNMP Configuration

Chapter 7

MIB Structure and SNMP 127

SNMP Overview	127
Definitions.....	127
SNMP Versions	128
SNMP Behaviour.....	129
SNMPv3 Special Behaviour	130
SNMP Configuration via a Configuration File.....	131
MIB Structure	135
Textual Conventions.....	136
Objects, Conformance, and Events.....	136
IP Addresses	137
Persistence.....	137
Changing a Parameter Value	137
Tables	138
Generic Variables	138
Variables for Administrative Commands	139
SNMP Access Limitation	140
SNMP Service Agent	140
Current MIB Version	140
Sending Configuration Data to the Mediatix 4100	141
Configuration File	141
Management Information Base – MIB	141

Chapter 8

IP Address and Network Configuration 143

IP Addresses	143
IP Addresses Formats in the DHCP Server	143
Provisioning Source	144

Services	144
Configuration Source	144
Local Host	145
Static DNS	147
Image	148
Management Server	149
Configuration File Fetching	150
Syslog	150
SIP Servers	151
SNTP	152
ETH2 Connector Static IP Address	153
DHCP Configuration	153
DHCP Options Waiting Time	153
Bootp BROADCAST Flag in DHCP Requests	154
Changing the Size of DHCP Requests	154
DHCP Server Configuration	155
Connection to the DHCP Behaviour	155
Network Configuration	155
Vendor and Site Specific DHCP Options	156
Vendor Specific Options	156
Site Specific Options	157
Option Codes	157
Entering IP Addresses	158
Entering FQDNs	159
Settings Example	161
Error Handling	161
DHCP Server Failures	162
Vendor/Site Specific Option Missing	162
DNS Failures	162
Ethernet Connection Speed	162
Speed and Duplex Detection Issues	163

Chapter 9

SIP Servers 165

Registrar Server	165
Configuration Source	165
Proxy Server	167
Configuration Source	167
Outbound Proxy Server	168
Configuration Source	169
Loose Router Configuration	170
Presence Compositor Server	171
Configuration Source	171

Chapter 10

DNS SRV Configuration..... 173

What is a DNS SRV?	173
Priority vs Weight	173
DNS SRV Call Flow	174
Enabling DNS SRV on the Mediatix 4100	174
DNS SRV Record Lock	175
DNS SRV-Oriented Settings	175

Chapter 11

Country-Specific Configuration	177
Caller ID Information	177
Caller ID Generation	177
ADSI	178
Setting the Location (Country)	178
Caller ID Selection	179
Custom Tone Configuration	180
Pattern Definition	180
Customizing the Tones	181
Custom Tone Example	182

Chapter 12

Configuration File Download	187
Configuration File Download Server	187
Configuring the TFTP Server	187
Configuring the SNTP Server	187
Configuring the HTTP Server	187
Configuring the HTTPS Server	188
Configuration File Server Settings	189
Setting up the Configuration File Download	190
Configuration Update Status	192
Configuration Files Encryption	193
Configuration Download via TFTP	194
Configuration Download via HTTP/HTTPS	195
Automatic Configuration Update	196
Error Handling	198
Management Server	201
Management Server Configuration	201
Downloading from the Management Server	201
Error Handling	202
Syslog Messages	203
Configuration File Example	204
Supported Characters	205

Chapter 13

Software Download	207
Before Downloading	207
Configuring the TFTP Server	207
Configuring the SNTP Server	207
Configuring the HTTP Server	207
Configuring the HTTPS Server	208
Software Servers Configuration	209
DHCP Configuration	209
Static Configuration	210
Download Procedure	211
Extracting the Zip File	211
Setting up the Image Path	211
Software Download Status	213
Download via TFTP	215
Download via HTTP/HTTPS	216
Automatic Software Update	217

Spanning Tree Protocol (STP)	219
Software Downgrade	220
Emergency Software Procedure	220
Using the Emergency Software	220

Chapter 14

Line Configuration 223

Lines Administrative State	223
Temporary Administrative State	223
Permanent Administrative State	224
Unregistered Line Behaviour	224
Flash Hook Detection	224
Source Line Selection	225
Examples of Source Line Selection Use	225
Loop Current	226
Callee Hang-up Supervision	227
Line Reversal	228
Blanking of an Anonymous Caller ID	228
Calling Number Transformation	229
Regular Expressions	229

Chapter 15

Voice Transmissions 231

Codec Descriptions	231
G.711 PCMA and PCMU	231
G.726	232
G.723.1	232
G.729	232
Preferred Codec	233
Enabling Individual Codecs	233
Packetization Time	234
DTMF Transport Type	237
DTMF Transport Using SIP INFO	238
DTMF Payload Type	238
DTMF – RFC 2833 Events	239
DTMF Transport over the SIP Protocol	239
DTMF Detection	240
DTMF Frequencies	240
DTMF Detection Configuration	241
Adaptative Jitter Buffer	243
About Changing Jitter Buffer Values	243
Voice Activity Detection	244
G.711 and G.726 VAD	244
G.729 VAD	244
Echo Cancellation	245
Signal Limiter	245
Comfort Noise	246
User Gain	246

Chapter 16

Fax Transmission	249
Introduction	249
Fax Calling Tone Detection	249
CED Fax Tone Detection	250
Analog CED Detection Behaviour	250
Clear Channel Fax	251
Data Codec Selection Procedure	253
T.38 Fax	254
T.38 No-Signal	255
T.38 INVITE Rejected with 606	256

Chapter 17

Bypass Configuration	257
Bypass Connector Settings	257
Standard Bypass	257

Chapter 18

SIP Protocol Features	259
User Agents	259
Home Domain Override	260
SIP User Agent Header	260
Session Timers	261
Session Timer Version	261
Background Information	262
Authentication	262
Line-Specific Authentication	262
Unit Authentication	263
Authentication Request Protection	263
SIP Trusted Sources	264
NAT Traversal	264
Mediatix 4100 Configuration	265
NAT System Configuration	265
SIP Transport Type	265
Transport Parameter	266
UDP Source Port Behaviour	266
SIP Penalty Box	267
Penalty Box vs Transport Types	267
Penalty Box Configuration	267
Registration Parameters	268
Refreshing Registration	268
Registration Expiration	268
Default Registration Expiration	269
Publication Parameters	269
Refreshing Publications	269
Publications Expiration	270
Default Publication Expiration	270
Interop Parameters	271
Call Transfer Capacity	271
Transmission Timeout	273
Max-Forwards Header	273

Referred-By Field	274
Direction Attributes in a Media Stream	274
Allowing Multiple Active Media in Answer	276
Local Ring Behaviour on Provisional Response	276
SIP Credential	277
Branch Parameter Settings	277
Ringing Response Code	279
URI-Parameters	279
Unsupported INFO Request	279
Outbound Proxy Usage	280
International Code Mapping	280
T.38 Negotiation Syntax	280
Addressing Failed Registration Attempts	281
SIP Domain in Request URI	281
SIP From: URI Content	281
Network Asserted Caller ID	282
Payload Type Settings	282
Controlling the Call Waiting Tone via SIP INFO	283
Ignore Username Parameter	283
Escaping the Pound (#) Character in SIP URI Username	284
SIP OPTIONS Method Support	284
Offer/Answer Model	284
Allow Media Reactivation in Answer	285
Allow Audio and Image Negotiation	285
Codec Order in Answer	286
UDP Checksum	286

Chapter 19

STUN Configuration 287

What is STUN?	287
SIP Outbound Proxy	287
Restrictions on the Media5 STUN Implementation	287
STUN Client Configuration	288

Chapter 20

SNTP Settings..... 289

Enabling the SNTP Client	289
Configuration Source	290
DHCP Configuration	290
Static Configuration	290
Defining a Custom Time Zone	291
STD / DST	291
OFFSET	291
START / END	291
Example	292

Chapter 21

Digit Maps 293

What is a Digit Map?	293
Syntax	293
Special Characters	294

How to Use a Digit Map	294
Combining Several Expressions	294
Using the # and * Characters	295
Using the Timer	295
Calls Outside the Country	295
Example	295
Validating a Digit Map	295
Processing Digits When Pressed	296
Setting up Digit Maps.....	296
Refused Digit Maps	297
Digit Maps Timeouts.....	298
Digit Map Examples.....	298
Digit Map Example 1 – Standard Calls.....	298
Digit Map Example 2 – PBX Emulation	300

Chapter 22

Telephony Features 303

Making Calls	303
Complete Dialing Sequence	303
Dialing a Telephone Number or Numerical Alias	303
Emergency Call	304

Chapter 23

Subscriber Services..... 305

Service Activation Processing	305
Call Hold	306
Enabling Call Hold.....	307
Using Call Hold.....	307
Second Call	307
Enabling Second Call	307
Using Second Call.....	307
Call Forward	308
Unconditional.....	308
On Busy.....	310
On No Answer	312
Call Waiting.....	314
Setting up Call Waiting	314
Using Call Waiting	315
Permanently Removing the Call Waiting Tone.....	316
Call Transfer	317
Blind Transfer	317
Attended Transfer.....	318
Conference Call	319
Requirements	319
Enabling the Conference Call Feature	320
Managing a Conference Call	321

Chapter 24

Telephony Attributes 323

Automatic Call.....	323
Call Direction Restriction	324

Hook Flash Processing	324
IP Address Call Service	325
Enabling IP Address Calls	325
Dialing an IP Address	325
PIN Dialing	326
Remote Line Extension	327
Delayed Hot Line	328
Call Rejection	329

Chapter 25

Message Waiting Indicator 331

What is Message Waiting Indicator (MWI)?	331
Standard MWI Methods	331
MWI Method #1	331
MWI Method #2	333
MWI Notify Service	333
How does the Service Work?	333
Configuring the IP Communication Server	334
Configuring the Mediatix 4100	334

Chapter 26

Management Server Configuration..... 335

Using the Management Server	335
Configuration Source	335

Chapter 27

Quality of Service (QoS) 337

Differentiated Services (DS) Field	337
IEEE 802.1q	338
Voice QoS vs RTCP Packets	339
VLAN	339

Chapter 28

Syslog Daemon 341

Syslog Daemon Configuration	341
Configuration Source	342
Customizing Syslog Messages	343
Configuring the Syslog Daemon Application	343

Chapter 29

Statistics 345

RTP Statistics	345
Statistics Buffers	345
How are Statistics Collected?	345
Statistics by Syslog	346
Example	347

Chapter 30

Maximum Transmission Unit (MTU)	349
What is MTU?	349
Mediatrix 4100 MTU	349
Possible Hardware Problem	349

Chapter 31

Troubleshooting	351
General Operation Issues	351
Calling Issues	354
Fax Issues	355
Tested Fax Models	356
Issues Arising from Specific Combinations/Scenarios	357
Configuration Issues	358
Software Upgrade Issues	358
SNMP Management Software Issues	360

Appendices

Appendix A

Standards Compliance and Safety Information	365
Standards Supported	365
Disclaimers	366
Federal Communications Commission (FCC) Part 15	366
Federal Communications Commission (FCC) Part 68	366
Industry Canada	367
CE Marking	367
RoHS China	368
Translated Warning Definition	369
Safety Warnings	370
Circuit Breaker (15A) Warning	370
TN Power Warning	370
Product Disposal Warning	370
No. 26 AWG Warning	370
LAN and FXS Ports Connectors Warning	370
Socket Outlet Warning	370
Safety Recommendations	371

Appendix B

Standard Hardware Information	373
Industry Standard Protocols	373
Hardware Features	373
Display	373
Interfaces	374
Power	374

Casing / Installation	374
Product Architecture Details	374
Real Time Fax Router Technical Specifications	374
Analog Line Interface (FXS)	375
Audio Specifications	375
DTMF Tone Detection	375
DTMF Tone Generation	376
MTBF Value	376
Power Consumption	376
Measurements at the DC input.....	376
Operating Environment.....	376
Dimensions and Weight.....	377
Warranty	377

Appendix C

Cabling Considerations 379

RJ-45 Cable	379
Straight Through Cable	379
Crossover Cable.....	380
RJ21X Connector (Mediatix 4124)	381
Creating a RJ21X Cable.....	381
RJ21X Pinout	381
RJ-11 (Telephone) Cable	383
Wiring Conventions	383
Loop Current vs Cable Length.....	384

Appendix D

Country-Specific Parameters 385

Definitions	385
Conventions	385
Distinctive Ring.....	387
Australia.....	388
Australia 1	388
Australia 2	389
Australia 3	390
Austria.....	391
Austria 1	391
Austria 2	391
Brazil	392
Chile.....	393
Chile 1	393
Chile 2	394
China	395
Czech Republic.....	396
Denmark	397
France.....	398
Germany.....	399
Germany 1.....	399
Germany 2.....	400
Germany 3.....	400
Hong Kong.....	401
Indonesia	402

Israel	403
Italy	404
Japan	405
Malaysia.....	406
Mexico	407
Netherlands	408
New Zealand.....	409
North America.....	410
North America 1	410
North America 2	411
Russia.....	412
South Africa	413
Spain.....	414
Sweden.....	415
Switzerland	416
Thailand	417
United Arab Emirates.....	418
UK.....	418

Appendix E

Glossary	419
-----------------------	------------

Appendix F

List of Acronyms	427
-------------------------------	------------

Appendix G

List of MIB Parameters	429
-------------------------------------	------------



About this Manual

Thank you for purchasing the Mediatrix 4100 Series from Media5.

The Mediatrix 4100 Series FXS VoIP Integrated Access Devices are telephony adaptors that connect up to 24 conventional analog telephones or fax machines to a LAN or a WAN with access to an IP Packet Network to permit high-quality, full duplex, audio/fax communications. The Mediatrix 4100 Series are available in the following models:

Table 1: Mediatrix 4100 Series Models

Model	Interfaces
Mediatrix 4108	8 FXS ports
Mediatrix 4116	16 FXS ports
Mediatrix 4124	24 FXS ports

To ensure maximum flexibility, the Mediatrix 4100 can:

- ▶ dynamically detect the most commonly used IP Telephony codecs and fax protocols, including T.38
- ▶ be auto-provisioned and remotely managed and upgraded
- ▶ provide a connection directly to the SCN via an automatic Bypass function in the event of an IP network failure or power outage.

Document Objectives

The *Mediatrix 4100 Reference Manual* provides technical information for the Mediatrix 4100.

Use the *Mediatrix 4100 Reference Manual* in conjunction with the appropriate publications listed in [“Related Documentation” on page xviii](#).



Note: There are many flavours of the Mediatrix 4100 unit. Because of this, some of the information provided may not apply to your particular Mediatrix 4100 unit model.

Intended Audience

This manual provides all the technical information needed to install and manage the Mediatrix 4100. It is intended for network administrators and system managers who install and set up network equipment; consequently, it assumes a basic working knowledge of LANs. Some installation procedures also require a qualified telephony technician. See [“RJ21X Connector \(Mediatrix 4124\)” on page 381](#) for more details.

From the perspective of the LAN administrator, a Mediatrix 4100 presents itself like another device to add to the LAN. It requires the same kind of TCP/IP addressing. The Mediatrix 4100 can also use a DHCP server on the LAN to automatically receive its IP configuration assignment.

Related Documentation

In addition to this manual, the Mediatrix 4100 document set includes the following:

- ▶ *MIB Reference Manual*
Lists and explains all parameters in the MIB structure.
- ▶ *Mediatrix 4100 Quick Start Guide*
This printed booklet allows you to quickly setup and work with the Mediatrix 4100.
- ▶ *Third Party Software Copyright Information*
This document lists the third-party software modules used in the Mediatrix 4100 along with any copyright and license information. This document is available at: http://www.media5corp.com/repository/common%20manuals/Third-Party_Software_Copyright_Information_Mediatrix.pdf.

Be sure to read any readme files, technical bulletins, or additional release notes for important information.

Document Structure

The Mediatrix 4100 Reference Manual has three parts:

- ▶ [“Installation and Web Page Configuration” on page 1](#). This part describes various installation of the Mediatrix 4100 and how to configure the unit via its web interface.
- ▶ [“SNMP Configuration” on page 125](#). This part describes all SNMP-related parameters of the Mediatrix 4100.
- ▶ [“Appendices” on page 363](#): This part contains supplemental information useful to the reader.

The Mediatrix 4100 Reference Manual contains the following information.

Table 2: Mediatrix 4100 Reference Manual Chapter/Appendices

Title	Summary
Installation and Web Page Configuration	
“Chapter 1 - Installation” on page 3	Describes the various installation scenarios of the Mediatrix 4100. Also presents the possible states and LED patterns of the Mediatrix 4100, as seen from an operator perspective.
“Chapter 2 - Web Interface – Introduction” on page 25	Describes how to access the embedded web server of the Mediatrix 4100 to set parameters by using the HTTP protocol.
“Chapter 3 - Web Interface – Management” on page 31	Describes the <i>Management</i> page of the web interface, which allows you to configure the configuration file download and firmware download parameters of the Mediatrix 4100.
“Chapter 4 - Web Interface – SIP Parameters” on page 63	Describes the <i>SIP</i> page of the web interface, which allows you to configure various SIP-related parameters of the Mediatrix 4100.
“Chapter 5 - Web Interface – Telephony” on page 73	Describes the <i>Telephony</i> page of the web interface, which allows you to configure the various telephony parameters of the Mediatrix 4100.
“Chapter 6 - Web Interface – Advanced” on page 117	Describes the <i>Advanced</i> page of the web interface, which allows you to configure various system and network parameters of the Mediatrix 4100.

Table 2: Mediatrix 4100 Reference Manual Chapter/Appendices (Continued)

Title	Summary
SNMP Configuration	
“Chapter 7 - MIB Structure and SNMP” on page 127	Describes how the Mediatrix 4100 uses the SNMP protocol for its configuration.
“Chapter 8 - IP Address and Network Configuration” on page 143	Describes how to set IP information in the Mediatrix 4100 and how to configure a DHCP server.
“Chapter 9 - SIP Servers” on page 165	Describes how to configure the Mediatrix 4100 to properly use the SIP servers.
“Chapter 10 - DNS SRV Configuration” on page 173	Describes the Mediatrix 4100's behaviour with a DNS SRV.
“Chapter 11 - Country-Specific Configuration” on page 177	Describes how to set the Mediatrix 4100 with the proper country settings.
“Chapter 12 - Configuration File Download” on page 187	Describes how to use the configuration file download feature to update the Mediatrix 4100 configuration.
“Chapter 13 - Software Download” on page 207	Describes how to download a software version available on the designated software server into the Mediatrix 4100.
“Chapter 14 - Line Configuration” on page 223	Describes the features available on the lines connected to the Mediatrix 4100.
“Chapter 15 - Voice Transmissions” on page 231	Describes the various codecs the Mediatrix 4100 supports for transmitting audio signals.
“Chapter 16 - Fax Transmission” on page 249	Describes how to perform fax transmissions in clear channel and T.38 with the Mediatrix 4100.
“Chapter 17 - Bypass Configuration” on page 257	Describes the Bypass feature that can be used in the event of a power failure or network failure.
“Chapter 18 - SIP Protocol Features” on page 259	Describes the SIP-specific feature to set up to properly use the SIP signalling programs and information defined in the Media5 SIP stack.
“Chapter 19 - STUN Configuration” on page 287	Describes how to configure the STUN client of the Mediatrix 4100.
“Chapter 20 - SNTP Settings” on page 289	Describes how to configure the Mediatrix 4100 to enable the notion of time (date, month, time) into it.
“Chapter 21 - Digit Maps” on page 293	Describes how to use a Digit Map to compare the number users dialed to a string of arguments.
“Chapter 22 - Telephony Features” on page 303	Explains how to perform basic calls with the Mediatrix 4100 and set the telephony variables of the unit to define the way it handles calls.
“Chapter 23 - Subscriber Services” on page 305	Describes how to set and use the subscriber services available on the user's telephone.
“Chapter 24 - Telephony Attributes” on page 323	Describes the telephony attributes available on the Mediatrix 4100.
“Chapter 25 - Message Waiting Indicator” on page 331	Explains how to set the Mediatrix 4100 to use the Message Waiting Indicator service.

Table 2: Mediatrix 4100 Reference Manual Chapter/Appendices (Continued)

Title	Summary
“Chapter 26 - Management Server Configuration” on page 335	Describes how to configure the Mediatrix 4100 to connect to a module or software that is used to remotely set up Mediatrix units.
“Chapter 27 - Quality of Service (QoS)” on page 337	Defines the QoS (Quality of Service) features available on the Mediatrix 4100.
“Chapter 28 - Syslog Daemon” on page 341	Describes how to configure and use the Syslog daemon.
“Chapter 29 - Statistics” on page 345	Defines the statistics the Mediatrix 4100 can collect.
“Chapter 30 - Maximum Transmission Unit (MTU)” on page 349	Describes the MTU (Maximum Transmission Unit) requirements of the Mediatrix 4100.
“Chapter 31 - Troubleshooting” on page 351	Examines some of the problems you may experience when connecting the Mediatrix 4100 to the network and provides possible solutions.
Appendices	
“Appendix A - Standards Compliance and Safety Information” on page 365	Lists the various standards compliance of the Mediatrix 4100.
“Appendix B - Standard Hardware Information” on page 373	Lists the technical hardware information of the Mediatrix 4100.
“Appendix C - Cabling Considerations” on page 379	Describes the pin-to-pin connections for cables used with the Mediatrix 4100.
“Appendix D - Country-Specific Parameters” on page 385	Lists the various parameters specific to a country such as loss plan, tones and rings, etc.

Document Conventions

The following information provides an explanation of the symbols that appear on the Mediatrix 4100 and in the documentation for the product.

Warning Definition



Warning: Means danger. You are in a situation that could cause bodily injury. Before you work on any equipment, you must be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents.

Where to find Translated Warning Definition

For safety and warning information, see [“Appendix A - Standards Compliance and Safety Information” on page 365](#).

This Appendix describes the international agency compliance and safety information for the Mediatrix 4100. It also includes a translation of the safety warning listed in the previous section.

Other Conventions

The following are other conventions you will encounter in this manual.



Caution: Caution indicates a potentially hazardous situation which, if not avoided, may result in minor or moderate injury and/or damage to the equipment or property.



Note: Note indicates important information about the current topic.

Standards Supported

Indicates which RFC, Draft or other standard document is supported for a specific feature.



This symbol indicates you can also set the current configuration by using the Unit Manager Network Graphical User Interface. The text will provide the location in the *Unit Manager Network Administration Manual* where to find information related to the specific configuration.

SCN vs PSTN

In Media5' and other vendor's documentation, the terms SCN and PSTN are used. A SCN (Switched Circuit Network) is a general term to designate a communication network in which any user may be connected to any other user through the use of message, circuit, or packet switching and control devices. The Public Switched Telephone Network (PSTN) or a Private Branch eXchange (PBX) are examples of SCNs.

Standards Supported

When available, this document lists the standards onto which features are based. These standards may be RFCs (Request for Comments), Internet-Drafts, or other standard documents.

The Mediatrix 4100's implementations are **based** on the standards, so it's possible that some behaviour differs from the official standards.

For more information on and a list of RFCs and Internet-Drafts, refer to the IETF web site at <http://www.ietf.org>.

Obtaining Documentation

These sections explain how to obtain documentation from Media5.

Media5 Web Site

Media5 offers the latest version of its products' documentation on its web site. You will thus be able to access and download the most current Media5 documentation. Follow this link: <http://www.media5corp.com/en/documentation>.



Note: This site does not contain any firmware versions.

Media5 Download Portal

Media5 offers online documentation via a self register web-portal. You will thus be able to access and download the most current Media5 documentation. Follow this link to register: <http://www.media5corp.com/en/support-portal>.



Note: This site does not contain any firmware versions.

Documentation Feedback

Media5 welcomes your evaluation of this manual and any suggestions you may have. These help us to improve the quality and usefulness of our publications.

Please send your comments to:

Media5 Corporation
Attention: Documentation Department
4229, Garlock Street
Sherbrooke, Quebec
Canada J1L 2C8
FAX: +1 (819) 829-5100

We appreciate your comments.

Unit Manager Network – Element Management System

The Unit Manager Network is a user-friendly element management system designed to facilitate the deployment, configuration and provisioning of Mediatrix access devices and gateways.

The Unit Manager Network offers the following key features, enabling the simple and remote configuration and deployment of numerous Mediatrix units:

- ▶ Detection of the state of each Mediatrix unit (e.g. power on/off).
- ▶ Automatic update of the list with installation of new Mediatrix units.
- ▶ Real-time graphical presentation of actual configuration.
- ▶ Tracking of all configuration options of the Mediatrix units on the network.
- ▶ Control of configuration parameters of all Mediatrix units within the same network.
- ▶ Storage of backup configuration file of each Mediatrix unit.
- ▶ Display of firmware release for any Mediatrix unit.
- ▶ Field-upgrade of all Mediatrix units.
- ▶ Controlled Implementation of new software.
- ▶ Supports SNMP requests: GET, GET NEXT, GET TABLE, GET WALK, SET, TRAP.
- ▶ SNMP abstraction layer: configuration can be changed without SNMP MIB knowledge.

The demo version of the Unit Manager Network is available on the Media5 Download Portal at: <https://support.mediatrix.com/DownloadPlus/Download.asp>.

See the *Unit Manager Network Administration Manual* for more details on how to use it to configure any Mediatrix 4100 unit on the network.

End User Technical Support

In order to maximize technical support resources, Media5 works through its partners to resolve technical support issues. All end users requiring technical support are encouraged to contact their vendor directly.

Installation and Web Page Configuration

Page Left Intentionally Blank

This chapter describes the installation and initial provisioning of the Mediatrix 4100.

Requirements

The Mediatrix 4100 requires the following items to work properly:

Table 3: Required Items

Item	Description
Analog lines (Mediatrix 4124 only)	Standard analog lines wired into a RJ21X cable. This cable may be directly connected to the Key Service Unit (KSU) or PBX Main Distribution Frame. See “RJ21X Connector (Mediatrix 4124)” on page 381 for more details.
DHCP Server (optional)	Supplies network parameters to the Mediatrix 4100.
DNS Server (optional)	Translates domain names into IP addresses.
SIP Server	Manages the active calls of the Mediatrix 4100.
Management Server (optional)	Module or software used to remotely manage and configure the Mediatrix 4100. Such software could be the Media5 Unit Manager Network. See “Unit Manager Network – Element Management System” on page xxii for more details.
TFTP Server or HTTP Server	Necessary for software updates.
Syslog Daemon (optional)	Receives all status messages coming from the Mediatrix 4100.

Safety Recommendations

To ensure general safety, follow these guidelines:

- ▶ Do not open or disassemble the Mediatrix 4100.
- ▶ Do not get the Mediatrix 4100 wet or pour liquids into it.
- ▶ Do not perform any action that creates a potential hazard to people or makes the equipment unsafe.

Package Contents

The Mediatrix 4100 package contains the following items:

- ▶ the Mediatrix 4100 unit
- ▶ two rack-mounting brackets
- ▶ a power cord for the country in which you are using the Mediatrix 4100
- ▶ a printed Flyer

You also need a 10/100 BaseT Ethernet RJ-45 cable.

Overview

The Mediatrix 4100 is a standalone Internet telephony access device that connects to virtually any business telephone system supporting standard analog lines.

The Mediatrix 4100 offers two Ethernet connectors switches enabling to establish two connections between conventional analog telephones or Group 3 fax machines and either a WAN, a LAN or a personal computer.

This version of the Mediatrix 4100 uses the Session Initiation Protocol (SIP), which is a protocol for transporting call setup, routing, authentication, and other feature messages to endpoints within the IP domain.

About the Mediatrix 4100

The Mediatrix 4100:

- ▶ Merges voice and data traffic onto a single unified network. Carrying telephone traffic over data networks uses less bandwidth (as compared to telephone trunks), resulting in a more cost-effective network solution.
- ▶ Easily integrates with existing telephone equipment. It converts any conventional analog telephone or fax machine into an Internet device.
- ▶ Bypasses long-distance toll charges for realized savings.
- ▶ Supports 10 Mb/s and 100 Mb/s Ethernet networks.
- ▶ Upgrades software easily for future enhancements.
- ▶ Uses the latest standards in Internet Telephony.
 - SIP protocol for call management
 - T.38 for fax relay
- ▶ Supports the following Codecs:
 - G.711 (μ -law, A-law)
 - G.723.1A
 - G.726
 - G.729 A
 - G.729 A rev. B
 - T.38 (fax) over UDP or TCP
- ▶ Supports Quality of Service technologies.
 - Differentiated Services (DS) Field
 - IEEE 802.1p user priority tagging
- ▶ Offers an intuitive Web-based management interface to simplify operation and support.

Placing a Call

You can place a call from a telephone or fax connected to:

- ▶ a port of the Mediatrix 4100 (Mediatrix 4108/4116)
- ▶ the analog line plugged into the Mediatrix 4100 (Mediatrix 4124)

The unit automatically detects if the call originates from a voice or fax transmission and acts accordingly.

When placing a call, the Mediatrix 4100 collects the DTMF digits dialed and sends a message to the Registrar Server. The Registrar Server sends back a list of contacts where the dialed number could be located.

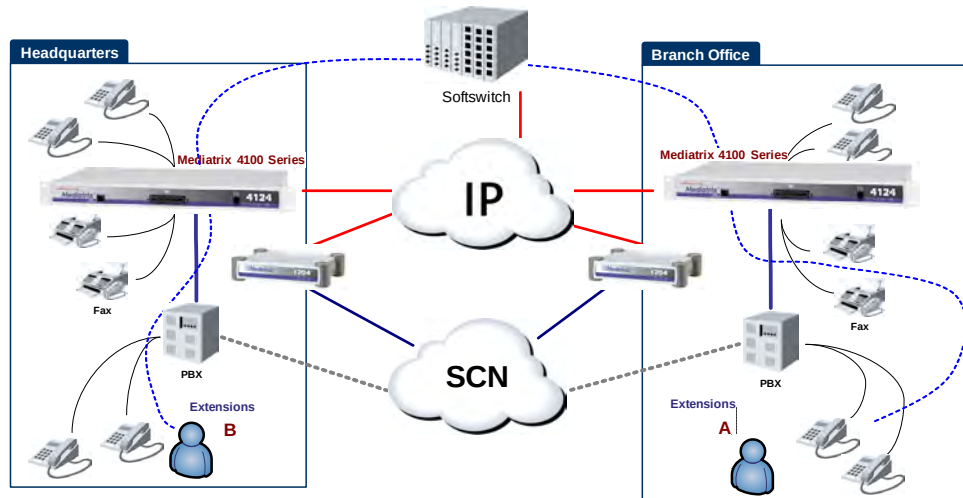
You can dial on a telephone/fax machine connected to the Mediatrix 4100 as you normally do.

Remote Office Connectivity Solution

The Mediatrix 4100 can be used to replace or complement existing PBX systems.

In the PBX trunk replacement scenario, the Mediatrix 4100 is directly connected to an existing PBX system via analog lines. Digital PBX phones are connected at the other end of the PBX. The Mediatrix 4100 allows you to send voice or fax communication on an IP network such as a LAN, WAN or the public Internet. You could still be connected to the standard SCN world by using either a Mediatrix 1204 analog gateway or your Internet Service Provider's gateway.

Figure 1: Remote Office Connectivity Solution

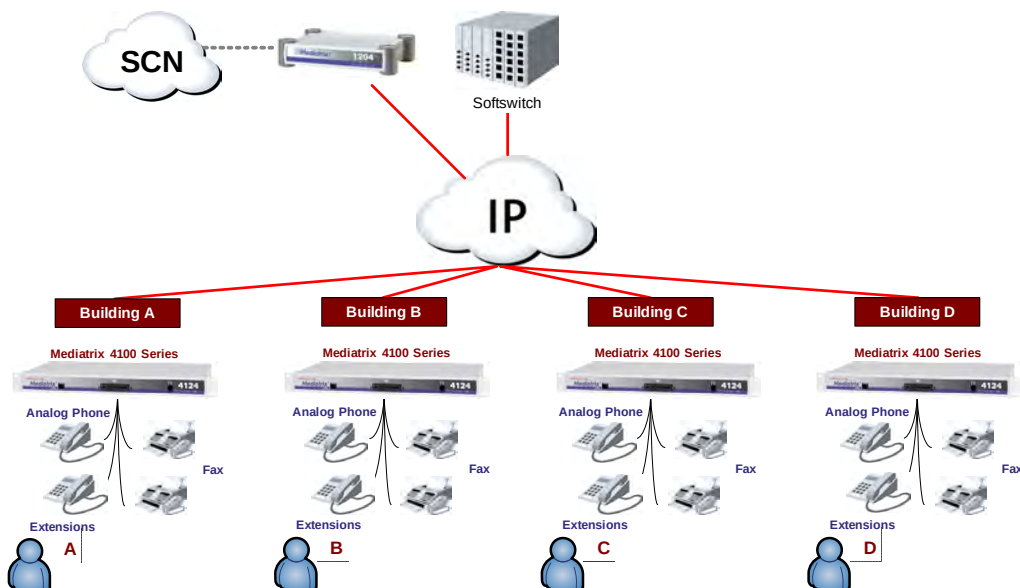


For more information on this scenario, please refer to *Configuration Notes 250: Configuring Mediatrix 41xx FXS Gateway and 1204 FXO Gateway in a Remote Office Environment* available on the Mediatrix Support Portal.

Multi-Tenant / Multi-Dwelling Units (MTU/MDU) IP Telephony

The Mediatrix 4100 can be used as a service unit for large buildings such as an apartment building. The analog lines of the building are directly connected to the Mediatrix 4100, which allows you to send voice or fax communication on an IP network such as a LAN, WAN or the public Internet.

Figure 2: MTU/MDU IP Telephony Solution



For more information on this scenario, please refer to *Configuration Notes 249: Configuring Mediatrix 41xx FXS Gateway with the Asterisk IP PBX System* available on the Mediatrix Support Portal.

Management Choices

The Mediatrix 4100 offers various management options to configure the unit.

Table 4: Management Options

Management Choice	Description
Web Interface	The Mediatrix 4100 web interface offers the following options: • Password-protected access via basic HTTP authentication, as described in RFC 2617 • User-friendly GUI Refer to “Chapter 2 - Web Interface – Introduction” on page 25 for more details.
SNMPv1/2/3	The Mediatrix 4100 SNMP feature offers the following options: • Password-protected access • Remote management • Simultaneous management Refer to “Chapter 7 - MIB Structure and SNMP” on page 127 for more details.
Auto-Update	The Mediatrix 4100 auto-update options are as follows: • Frequent polling • Automatic software and configuration files downloads • Configuration file encryption Refer to “Chapter 13 - Software Download” on page 207 and “Chapter 12 - Configuration File Download” on page 187 for more details.

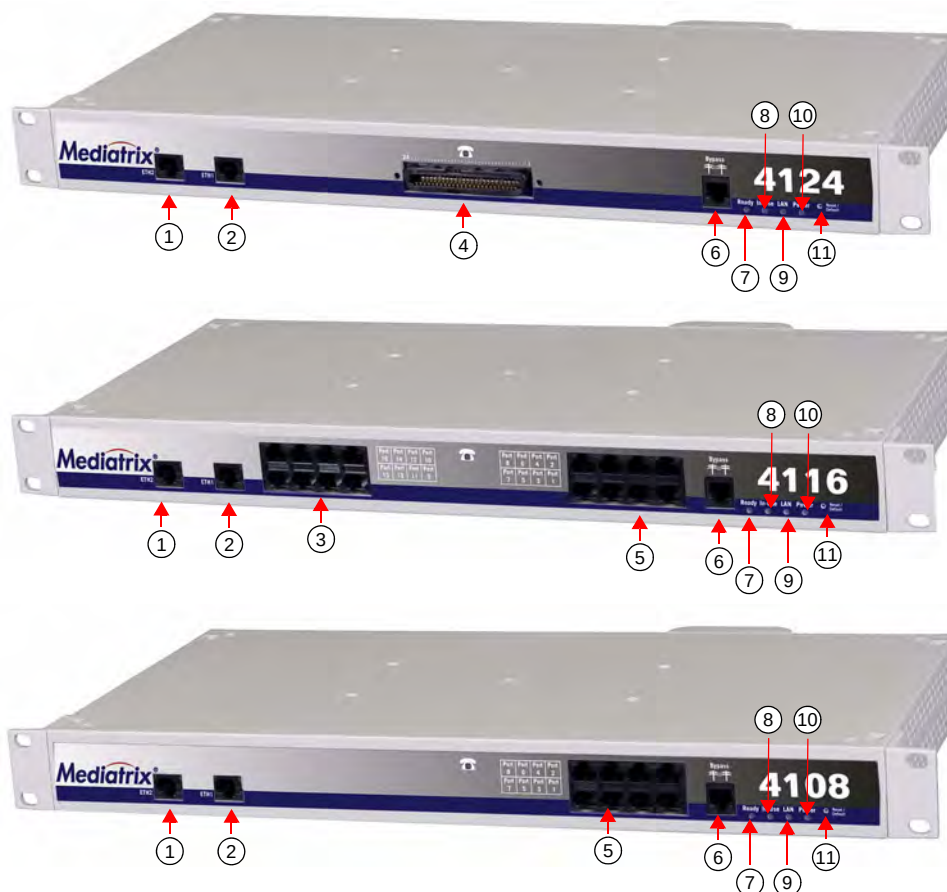
Panels

This section provides an overview of the front and rear panels of the Mediatrix 4100.

Front Indicators

[Figure 3](#) shows the visual indicators and various connectors located on the front of the Mediatrix 4100.

Figure 3: Front Panel Indicators and Connectors



[Table 5](#) describes the LEDs and connectors on the front panel of the Mediatrix 4100.

Table 5: Front Connectors and LEDs

	Connector / LED	Description
1.	ETH2	A 10/100 BaseT Ethernet RJ-45 connector that can be connected into the network card of a computer. This connector may not be available on older models.
2.	ETH1	RJ-45 connector for 100 BaseT Ethernet access to a LAN.
3.	FXS connectors	Eight RJ-11 connectors to attach conventional telephones or G3 fax machines. These connectors are available on the Mediatrix 4116 only.
4.	Analog Lines	A RJ21X connector with 24 RJ-11 Teladapt connectors to provide line cord connection to analog phones, faxes and modems. This connector is available on the Mediatrix 4124 only. See "RJ21X Connector (Mediatrix 4124)" on page 381 for more details.

Table 5: Front Connectors and LEDs (Continued)

	Connector / LED	Description
5.	FXS connectors	Eight RJ-11 connectors to attach conventional telephones or G3 fax machines. These connectors are available on the Mediatrix 4108 and Mediatrix 4116 only.
6.	Bypass	Permits users to make and receive calls even when the Mediatrix 4100 is not operating.
7.	Ready LED	When lit, the Mediatrix 4100 is ready to initiate or receive a call. The unit does not have to be registered to a server.
8.	In Use LED	When lit, at least one of the FXS lines is in use.
9.	LAN LED	Provides the state of the network connected to the <i>ETH1</i> and <i>ETH2</i> connectors
10.	Power LED	When lit, power is applied to the Mediatrix 4100.
11.	Reset / Default button	Resets configuration parameters of the Mediatrix 4100 to default (known) values. It can be used to reconfigure the unit. See "Reset / Default Switch" on page 19 for more details.

See ["LED Indicators" on page 15](#) for a detailed description of the LED patterns the Mediatrix 4100 may have and the states they represent.

Port Numbering Convention (Mediatrix 4108/4116)

The following describes the port numbering conventions of the FXS connectors available on the Mediatrix 4108 and Mediatrix 4116. Note that the connectors in the left section are available only on the Mediatrix 4116 model.

Figure 4: FXS Connectors Port Numbering Convention (Mediatrix 4108/4116)

Analog Lines Numbering Convention (Mediatrix 4124)

The following describes the numbering conventions of the FXS analog lines available on the Mediatrix 4124.

Figure 5: Analog Lines Numbering Convention (Mediatrix 4124)

Rear Connectors

The Mediatrix 4100 has several connections that must be properly set. [Figure 6](#) shows the back panel of the Mediatrix 4100.

Figure 6: Back Panel Connector



[Table 6](#) describes the back panel connections.

Table 6: Back Connections of the Mediatrix 4100

Connection	Description
Power connector	IEC320-C14 Power Cord Receptacle. This power cord supplies an internal Switching Mode Power Supply (SMPS) Universal input. The power supply range is 100 – 240 VAC; 50/60 Hz; 1.6 A.

Choosing a Suitable Installation Site



Warning: The analog lines of the Mediatrix 4100 are not intended for connection to a telecommunication network that uses outside cable.



Warning: To prevent fire or shock hazard do not expose the unit to rain or moisture.

The Mediatrix 4100 is suitable for use in an office environment where it can be placed in the same room or cabinet where the PBX/telephony equipment is located. The unit can be mounted on a standard 48.26 cm (19 in.) equipment rack.

Location

Install the Mediatrix 4100 in a well-ventilated location where it will not be exposed to high temperature or humidity. Do not install the Mediatrix 4100 in a location exposed to direct sunlight or near stoves or radiators. Excessive heat could damage the internal components.

When deciding where to position the Mediatrix 4100, ensure that:

- ▶ The Mediatrix 4100 is accessible and cables can be easily connected.
- ▶ The Mediatrix 4100 is installed in a restricted access location.
- ▶ The Mediatrix 4100 must be installed in a location where equipotential bonding has been applied in order to have all earthed equipments in the location at the same Earth potential .
- ▶ The cabling is away from the following:
 - Sources of electrical noise such as radios, transmitters, and broadband amplifiers.
 - Power lines and fluorescent lighting fixtures.
 - Water or moisture that could enter the casing of the Mediatrix 4100.
- ▶ The airflow is not restricted around the Mediatrix 4100 or through the vents in the side and back

of the unit. The unit requires a minimum of 25 mm (1 in.) clearance.

- ▶ The fan on the Mediatrix 4100 is not blocked or that the back of the unit is not too close to the wall.
- ▶ The operating temperature is between 0°C and 40°C.
- ▶ The humidity is not over 85% and is non-condensing.

Rack-Mounting

The Mediatrix 4100 is 4.4 cm (1.74 in.) high. It meets the EIA-310-D and ETS 300 119 standards.

▶ To rack-mount the Mediatrix 4100:

1. Disconnect all of the cables from the Mediatrix 4100.
2. Insert the Mediatrix 4100 into the rack and secure with suitable screws (not provided). Be sure that the ventilation holes are not obstructed.
3. Proceed to [“Hardware Connection” on page 10](#).

Condensation

When bringing the unit into a warm environment from the cold, condensation may result that might be harmful to the unit. If this occurs, allow the unit to acclimatize for an hour before powering it on.

Cleaning

To clean the Mediatrix 4100, wipe with a soft dry cloth. Do not use volatile liquids such as benzine and thinner that are harmful to the unit casing.

For resistant markings, wet a cloth with a mild detergent, wring well and then wipe off. Use a dry cloth to dry the surface.

Hardware Connection

This section describes how to set the connectors of the Mediatrix 4100.



Warning: Do not connect the Mediatrix 4100 directly to Telecommunication Systems.



Caution: The Mediatrix 4100 must be installed on a circuit equipped with a breaker so that you can easily power the unit off if required.

See [“Appendix C - Cabling Considerations” on page 379](#) for more details on the cables the Mediatrix 4100 uses.

Reserving an IP Address

Before connecting the Mediatrix 4100 to the network, Media5 strongly recommends that you reserve an IP address in your DHCP server – if you are using one – for the unit you are about to connect. This way, you know the IP address associated with a particular unit.

DHCP servers generally allocate a range of IP addresses for use on a network and reserve IP addresses for specific devices using a unique identifier for each device. The Mediatrix 4100 unique identifier is its media access control (MAC) address. You can locate the MAC address as follows:

- ▶ It is printed on the label located on the bottom side of the unit.
- ▶ It is located in the `sysMgmtMIB` under the `sysMacAddress` variable.

- You can dial the following digits on a telephone connected to the Mediatrix 4100:

***1

The Mediatrix 4100 answers back with its MAC address. See [“Special Vocal Features” on page 15](#) for more details.

Connecting Cables – Mediatrix 4108/4116 Models

The following describes how to connect the various cables to the Mediatrix 4108 or Mediatrix 4116.

Figure 7: Mediatrix 4108/4116 Hardware Installation



► To install the cables:

1. Before you begin, be sure that the Mediatrix 4100 is powered off.
2. Connect the power cord to its connector on the back of the unit.
3. Connect analog telephones or fax machines into the phone/fax connectors
The Mediatrix 4108 has 8 connectors, while the Mediatrix has 16 connectors.
Use a standard telecommunication cord with a minimum of 26 AWG wire size.



Note: The Mediatrix 4100 telephone line interface has been designed to interface with a conventional telephone or fax machine. Connections to FXS ports of third party devices such as a PBX / Key System could damage the Mediatrix 4100.

4. Connect a 10/100 BaseT Ethernet RJ-45 cable into the *ETH1* connector of the Mediatrix 4100; connect the other end to a compatible Ethernet interface that supplies TCP/IP network access (e.g., router, switch, hub or computer).
Use a standard telecommunication cord with a minimum of 26 AWG wire size.
5. Connect a 10/100 BaseT Ethernet RJ-45 cable into the *ETH2* connector of the Mediatrix 4100 and connect the other end to the network card of a computer.
This connector may not be available on older models.
Use a standard telecommunication cord with a minimum of 26 AWG wire size. You can either use a crossover or straight Ethernet cable because it performs automatic MDI / MDIX detection. See [“RJ-45 Cable” on page 379](#) for more details.
6. Connect a SCN line into the *Bypass* connector of the Mediatrix 4100 (optional).
Use a standard telecommunication cord with a minimum of 26 AWG wire size.

7. Connect the other end of the power cord to an electrical earthed socket-outlet.



Warning: The electrical outlet must be installed near the Mediatrix 4100 so that it is easily accessible.

Connecting Cables – Mediatrix 4124 Model

The following describes how to connect the various cables to the Mediatrix 4124.

Figure 8: Mediatrix 4124 Hardware Installation



► To install the cables:

1. Before you begin, be sure that the Mediatrix 4124 is powered off.
2. Connect the power cord to its connector on the back of the unit.
3. Connect a RJ21X cable into the RJ21X receptacle of the Mediatrix 4124.
The RJ21X cable contains the connections for the 24 analog lines to connect to the Mediatrix 4124. See ["RJ21X Connector \(Mediatrix 4124\)" on page 381](#) for more details.
4. Connect a 10/100 BaseT Ethernet RJ-45 cable into the *ETH1* connector of the Mediatrix 4124; connect the other end to a compatible Ethernet interface that supplies TCP/IP network access (e.g., router, switch, hub or computer).
Use a standard telecommunication cord with a minimum of 26 AWG wire size.
5. Connect a 10/100 BaseT Ethernet RJ-45 cable into the *ETH2* connector of the Mediatrix 4100 and connect the other end to the network card of a computer.
This connector may not be available on older models.
Use a standard telecommunication cord with a minimum of 26 AWG wire size. You can either use a crossover or straight Ethernet cable because it performs automatic MDI / MDIX detection. See ["RJ-45 Cable" on page 379](#) for more details.
6. Connect a SCN line into the *Bypass* connector of the Mediatrix 4124 (optional).
Use a standard telecommunication cord with a minimum of 26 AWG wire size.
7. Connect the other end of the power cord to an electrical earthed socket-outlet.



Warning: The electrical outlet must be installed near the Mediatrix 4124 so that it is easily accessible.

Starting the Mediatrix 4100 for the First Time

The default MIB parameters are set so that the unit can be directly plugged into a network and provisioned with a DHCP server. Media5 strongly recommends to set your DHCP server before installing the unit on the network. See [“Chapter 8 - IP Address and Network Configuration” on page 143](#) for more details.

If you are experiencing problems, or if you do not want to use a DHCP server, perform a Recovery Mode procedure, as explained in [“Recovery Mode” on page 20](#).

IP Address Discovery or Configuration



Note: This section applies to the IP configuration of the **ETH1** port.

Once the physical connection is complete and the Mediatrix 4100 is powered up, the first thing to do is find out the IP address the Mediatrix 4100 is using. The Mediatrix 4100's IP address can be set either dynamically or statically. The default behaviour of the Mediatrix 4100 is to try to obtain a dynamic IP address through a DHCP server.



Caution: If you set a Mediatrix 4100 with a static *eth1-4* IP address in a subnet (for instance, 192.168.200.1) and the *eth5* interface receives a dynamic IP address in the same subnet (via a DHCP server or PPP peer), you will not be able to contact the unit via the WAN. You must be careful that a dynamic IP address does not overlap a static IP subnet that is already configured. Note that the current default value of the Mediatrix 4100 is 192.168.0.10.

Dynamic IP Address Discovery

Before connecting the Mediatrix 4100 to the network, Media5 strongly suggests that you reserve an IP address in your DHCP server for the unit you are about to connect ([“Reserving an IP Address” on page 10](#)).

If you have not reserved an IP address, you can discover which IP address has been assigned to the Mediatrix 4100 by either:

- ▶ consulting your DHCP server's logs to find out details on the DHCP lease that was given to the Mediatrix 4100.
- ▶ using a network packet sniffer (e.g., Wireshark) to examine the DHCP messages exchanged between the Mediatrix 4100 and your DHCP server while the Mediatrix 4100 boots up.

▶ To start the Mediatrix 4100 with a dynamic IP address:

1. If you need to discover the IP address of the Mediatrix 4100, install and start your network packet sniffer.
2. Power on the Mediatrix 4100 by connecting the other end of the power cord to an electrical earthed socket-outlet. The electrical outlet must be installed near the Mediatrix 4100 so that it is easily accessible.



Note: If the *Power* LED is always blinking and never turns on, this means that the Mediatrix 4100 cannot find a DHCP server. Check that you have a DHCP server properly configured on your network. If you do not have a DHCP server, go to the section [“Default Static IP Address Configuration” on page 14](#).

Default Static IP Address Configuration

If there is no DHCP server in your network, then the IP address has to be configured statically.

► To start the Mediatrix 4100 with a static IP address:

1. With a 10/100 Hub and two 10/100 BaseT Ethernet RJ-45 straight cables, connect both cables to the Hub; one of them is connected into the *ETH1* connector of the Mediatrix 4100 and the other one links the computer to the Hub.
2. Reconfigure the IP address of your computer to *192.168.0.10* and the Subnet Mask to *255.255.255.0*. Restart the computer.
3. Power on the Mediatrix 4100 by connecting the other end of the power cord to an electrical earthed socket-outlet. The electrical outlet must be installed near the Mediatrix 4100 so that it is easily accessible.
4. Insert a small, unbent paper clip into the *Reset / Default* switch hole located at the front of the Mediatrix 4100. The *Power* LED will start blinking, and after a few seconds, all the LEDs will start blinking. Release the paper clip after all the LEDs start blinking and before they all stop blinking (between 5-10 seconds). Only the *Power* and *Ready* LEDs should go on blinking to inform you that the recovery mode procedure has been performed.

After a recovery mode is performed, the Mediatrix 4100 uses the default IP address *192.168.0.1*. Refer to ["Recovery Mode" on page 20](#) for details on the recovery mode procedure.

You must perform the recovery mode in a closed network and perform it on only one Mediatrix 4100 at a time, since the default IP address is the same on every unit.

Initial Provisioning Sequence

When starting the Mediatrix 4100 for the first time, it needs to be configured before it can support calls. This process is known as *provisioning*. This sequence assumes that you have installed the Mediatrix 4100 hardware as per ["Hardware Connection" on page 10](#).

The Mediatrix 4100 requests its configuration only on the first restart. You can change the configuration at will after the initial provisioning and the provisioning system can refresh the Mediatrix 4100 configuration. The provisioning system consists of the Management Server and a DHCP server. The Management Server includes a provisioning client, provisioning server, and SNMP proxy server.

► Initial provisioning sequence:

1. When the Mediatrix 4100 starts, it broadcasts a message requesting DHCP services (if the unit is configured to start in DHCP mode).
2. The DHCP server responds with a set of IP addresses and network parameters, one of which is the Mediatrix 4100 IP address.

The following are some of the network parameters assigned via DHCP:

- Mediatrix 4100 IP address
 - Subnet Mask
 - Default Router IP address
 - Primary and Secondary DNS IP addresses
 - Management Server IP address and port number (optional)
 - Configuration file server IP address and port number (optional)
 - SIP Servers IP address and port number
3. The Mediatrix 4100 may request its configuration in two ways:
 - by using the IP address of the Management Server to request its configuration.
 - by using a configuration file.

Special Vocal Features

When entering special characters on your telephone pad, the Mediatrix 4100 talks back to you with relevant information.

► **To access special vocal features:**

1. Take one of the telephones connected to the Mediatrix 4100.
2. Dial one of the digits sequence on the keypad.

Table 7: Special Vocal Features

Digits to Dial	Information Vocally Sent by the Mediatrix 4100
*#*0	Current IP address of the Mediatrix 4100 (static or DHCP).
*#*1	MAC address of the Mediatrix 4100.

LED Behaviour in Starting Mode

When the Mediatrix 4100 starts and it is not configured to use a DHCP server, it uses static IP addresses. If the static information is not valid, the *LAN* LED blinks at 1 Hz with 75% duty cycle. This lets you know that you must perform a Factory reset or Recovery mode operation. See [“Reset / Default Switch” on page 19](#) for more details.

LED Indicators

A LED can be ON, OFF, BLINKING or controlled by hardware (HW). The blinking behaviour is described in terms of rate (in Hertz – Hz) and duty cycle (in percentage). For instance, a LED that turns on every two seconds and stays on for one second would be described as: blink 0.5 Hz 50%. The hardware (HW) behaviour is not defined. It is usually the standard state for the *LAN* LED.

Ready LED

The *Ready* LED provides an “at-a-glance” view of the Mediatrix 4100 operational status. It is an aid for installation and on-site support. This LED is:

- ON when all elements of the *ifAdminOpState* column are “enabled”.
- OFF when all elements of the *ifAdminOpState* column are “disabled”.
- Blinking when at least one element of the *ifAdminOp State* column is “enabled” and at least one element is “disabled”.

Patterns and meanings of the *Ready* LED are described in [Table 10 on page 17](#).

Refer to the *MIB Reference Manual* for more details on the *ifAdminOpState* variable.

In Use LED

The *In Use* LED provides feedback of the activity on the line. If a line is ringing, off-hook, or displaying information (ADSI), then this LED is ON. The *In Use* LED is ON when at least one element in the *ifAdminUsageState* column is “busy”. Patterns and meanings of the *In Use* LED are described in [Table 10 on page 17](#).

Refer to the *MIB Reference Manual* for more details on the *ifAdminState* variable.

LAN LED

The *LAN* LED provides the Link and Heartbeat status of the network connected to the *ETH1* and *ETH2* connectors. If there is no link under HW control, the LED is OFF. When a link is established, but no activity is detected, the LED is ON; it turns off for very short periods of time when activity is detected and blinks rapidly when the LAN is loaded. Patterns and meanings of the *LAN* LED are described in [Table 10 on page 17](#).

Power LED

The *Power* LED indicates whether or not the Mediatrix 4100 is operational at its most basic level. It does not imply that the unit can be used, only that it is capable of being used. Healthy operation would be steady ON. Patterns and meanings of the *Power* LED are shown in [Table 10 on page 17](#).

LED Patterns

[Table 8](#) describes the different states a Mediatrix unit can have and their associated LED patterns.

Table 8: States and LED Patterns

State	Description	LEDs Pattern			
		Ready	In Use	LAN	Power
Booting	Follows a hardware start or a reset.	See "Booting LED Pattern Description" on page 17			
Normal Mode	"Normal" state of the unit where calls can be initiated. Each LED has a separate behaviour.	See "NormalMode LED Pattern Description" on page 17			
AdminMode	Calls are not permitted and maintenance actions can be performed.	See "AdminMode LED Pattern Description" on page 18			
Recovery Mode	The IP addresses for local host, image server, syslog server, etc., are temporarily set to known values. Calls are not allowed.	Blink 1 Hz 75%	Off	Misc. ^a	Blink 1 Hz 75%
Reset Pending	Triggered when the <i>Reset / Default</i> switch is pressed and held for at least 2 seconds.	Off	Off	Off	Blink 1 Hz 50%
Reboot Pending	Triggered when the <i>Reset / Default</i> switch is pressed in either the <i>ResetPending</i> or <i>RecoveryMode Pending</i> states. The unit prepares for a physical shutdown and restart.	Off	Off	Off	Off
Recovery Mode Pending	Triggered when the <i>Reset / Default</i> switch is held at start-time or for at least 7 seconds.	Blink 1 Hz 50%	Blink 1 Hz 50%	Blink 1 Hz 50%	Blink 1 Hz 50%
Default Settings Pending	Triggered when the <i>Reset / Default</i> switch is not released while in <i>RecoveryModePending</i> state. At run time, if the <i>Reset / Default</i> switch is released within 5 seconds, the unit applies default settings, otherwise the action is cancelled and the unit goes back to the Operation Modes state or it resets. At start time, the unit stays in this state until the <i>Reset / Default</i> switch is released. The unit then applies the default settings and restarts.	On	On	On	On
Image DownloadIn Progress	A software image is downloaded into the unit and written to persistent storage.	LEDs are blinking at 1 Hz 75%, <i>one at a time</i> , from left to right.			
Image Download Error	Triggered after a failure of an image download operation. After 4 seconds, the unit restarts.	Blink 2 Hz 50%	Blink 2 Hz 50%	Blink 2 Hz 50%	Blink 2 Hz 50%

Table 8: States and LED Patterns (Continued)

State	Description	LEDs Pattern			
		Ready	In Use	LAN	Power
InitFailed	Triggered when bad initialization parameters are detected and the unit cannot start correctly. Note: If the network configuration is dynamic, the unit stays in the <i>Booting</i> state and continues to query the DHCP until it receives valid values. If the configuration is static, the LED pattern indicates that the unit must be reset to default settings or put into recovery mode for maintenance and correction of network values.	Off	Off	Blink 4 Hz 50%	Off
DiagFailed	Triggered at start-time when the hardware or software diagnostic fails. This is a critical error and the unit may require RMA.	Off	Off	Off	Blink 4 Hz 50%
NetworkRescue	The unit tries to download and install a firmware given by the Network Rescue server.	Off	LEDs are blinking to show a LED displacing light from left to right and right to left.		

a. See the corresponding LED pattern in ["NormalMode LED Pattern Description" on page 17](#).

Booting LED Pattern Description

While in the *Booting* state, the LEDs of the Mediatrix 4100 behave independently; the following table indicates the behaviour for each LED.

Table 9: LED Patterns in Booting Mode

LED	Pattern	Meaning
Ready	Steady Off	Not Ready.
In Use	Steady Off	Cannot be in use.
LAN (HW Ctrl)	Steady On	Ethernet connection detected.
	Steady Off	Ethernet connection not detected or hardware control not activated.
	Blinking (variable rate)	Ethernet activity detected.
Power	Steady On	Power is On.
	Blinking 1 Hz 75%	Waiting for a DHCP answer.

NormalMode LED Pattern Description

While in the *NormalMode* state, the LEDs of the Mediatrix 4100 behave independently; the following table indicates the behaviour for each LED.

Table 10: LED Patterns in Operation Mode

LED	Pattern	Meaning
Ready	Steady On	All lines are enabled (operational state).
	Steady Off	All lines are disabled (operational state).
	Blink 0.25 Hz 75%	At least one line is enabled and at least one line is disabled (operational state).

Table 10: LED Patterns in Operation Mode (Continued)

LED	Pattern	Meaning
In Use	Steady On	At least one line is busy (usage state).
	Steady Off	All lines are not busy (usage state) or the unit is not connected to the network.
LAN (HW Ctrl)	Steady On	Ethernet connection detected.
	Steady Off	Ethernet connection not detected.
	Blinking (variable rate)	Ethernet activity detected.
Power	Steady On	Power is On.
	Steady Off	Power is Off.
	Blinking 1 Hz 75%	Waiting for a DHCP answer.

AdminMode LED Pattern Description

While in the *AdminMode* state, the LEDs of the Mediatrix 4100 behave independently; the following table indicates the behaviour for each LED.

Table 11: LED Patterns in AdminMode

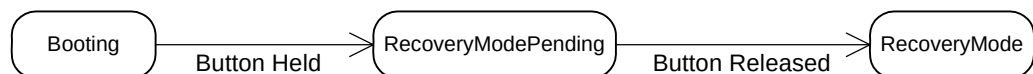
LED	Pattern	Meaning
Ready	Blinking 1 Hz 75%	<i>Ready</i> and <i>Power</i> LEDs blink off phase at 180 degrees.
In Use	Steady Off	All analog lines are not available.
LAN (HW Ctrl)	Steady On	Ethernet connection detected.
	Steady Off	Ethernet connection not detected.
	Blinking (variable rate)	Ethernet activity detected.
Power	Blinking 1 Hz 75%	<i>Ready</i> and <i>Power</i> LEDs blink off phase at 180 degrees.

Recovery Mode LED Patterns

There are two different sequences of LED patterns indicating that a recovery is in process.

At Start-Time

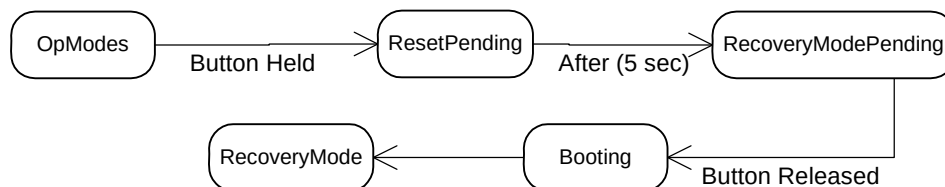
When pressing the *Reset / Default* switch at start-time, the state sequence goes as follows:

Figure 9: LED Pattern at Start-Time

At Run-Time

When pressing the *Reset / Default* switch at run-time, the state sequence goes as follows:

Figure 10: LED Patterns at Run-Time



Reset / Default Switch

The *Reset / Default* switch allows you to:

- ▶ Cancel an action that was started.
- ▶ Revert to known factory settings if the Mediatix 4100 refuses to work properly for any reason or the connection to the network is lost.
- ▶ Reconfigure a unit.

At Run-Time

The *Reset / Default* switch can be used at run-time – you can press the switch while the Mediatix 4100 is running without powering the unit off. [Table 12](#) describes the actions you can perform in this case.

Table 12: Reset / Default Switch Interaction

Reset / Default Switch Pressed for:	Action	Comments	LEDs Pattern			
			Ready	In Use	LAN	Power
2 to 5 seconds	Restarts the Mediatix 4100	No changes are made to the Mediatix 4100 settings.	Off	Off	Off	Blink
5 to 10 seconds	Restarts the Mediatix 4100 in Recovery Mode	Sets the Mediatix 4100 IP address to its default value in the MIB and restarts the unit.	Blink	Blink	Blink	Blink ^a
10 to 15 seconds ^b	Restarts the Mediatix 4100 in Factory Reset	Deletes the persistent MIB values, creates a new configuration file with the default factory values, and then restarts the unit.	On	On	On	On

a. Synchronized blinking at 2 Hz (50% duty cycle).

b. You can disable the Factory reset procedure to avoid users deleting the existing configuration. See ["Disabling the Factory Reset" on page 22](#) for more details.

At Start-Time

The *Reset / Default* switch can be used at start-time – you power the unit off, and then depress the *Reset / Default* switch and power the unit back on. In this case, the following explains the reset behaviour:

- ▶ Pressing the *Reset / Default* switch at startup until all the LEDs start blinking restarts the Mediatix 4100 in "Recovery Mode". Note that this can take up to 40 seconds.

- ▶ Pressing the *Reset / Default* switch at startup until all the LEDs stop blinking and remain ON applies the “Factory Reset” procedure. This feature reverts the Mediatrix 4100 back to its default factory settings.

See [“LED Indicators” on page 15](#) for a detailed description of the LED patterns related to the *Reset / Default* switch.

Recovery Mode

The recovery mode restarts the Mediatrix 4100 in a known, static, and minimal state. It is used to recover from a basic configuration error that prevents you to reach the unit through the network. It may serve as a last resort before the Factory reset command. You must perform it in a closed network and on only one Mediatrix 4100 at a time, because the default IP address is the same on every unit.

The recovery mode is not intended to address configuration and/or software problems. For those types of problems, you must use the Factory reset.



Note: The procedure below assumes that you are performing it at run-time.

▶ To trigger the Recovery Mode:

1. With a 10/100 Hub and two 10/100 BaseT Ethernet RJ-45 straight cables, connect both cables to the hub; one of them is connected into the *ETH1* connector of the Mediatrix 4100 and the other one links the computer to the hub.
Alternatively, you can connect a 10/100 BaseT Ethernet RJ-45 crossover cable into the *ETH1* connector of the Mediatrix 4100 and connect the other end to your computer.
2. Reconfigure the IP address of your computer to *192.168.0.10* and enter the Subnet Mask of *255.255.255.0*. Restart the computer.
3. Insert a small, unbent paper clip into the *Reset / Default* switch hole located at the front of the Mediatrix 4100.
4. Hold the *Reset / Default* switch between 5 and 10 seconds – until the LEDs start blinking.
5. Release the paper clip.

Only the *Power* and *Ready* LEDs should go on blinking to inform you that the recovery reset has been performed.

In recovery mode, the provisioning source of the *localHostConfigSource* variable is set to **default**, meaning that the default factory setting is used.

The following variables use their default values in the MIBs:

- *localHostAddress*
- *localHostPrimaryDns*
- *localHostSecondaryDns*
- *localHostDefaultRouter*
- *localHostSnmpPort*
- *localHostSubnetMask*
- *imagePrimaryHost*
- *imagePrimaryPort*
- *imageSecondaryHost*
- *imageSecondaryPort*
- *msHost*
- *msTrapPort*
- *syslogHost*
- *syslogPort*

The following variables of the *mediatrixMgmt* group are all set to static:

- *imageConfigSource*

- configFileFetchingConfigSource
- msConfigSource
- syslogConfigSource
- snmpConfigSource

All the persistent MIB values are kept.

In this mode, SIP is deactivated. Only SNMP or HTTP can be used to set the IP addresses listed above and the protocol-specific IP addresses (all IP addresses located under the *ipAddressConfig* folder in the MIB structure).

You can also download a software version, but you cannot download a configuration file.

6. When the Mediatrix 4100 has finished its provisioning sequence, perform the changes, and then turn it off, plug it on the network, and turn it on again.

When restarting, the Mediatrix 4100 will not be in Recovery mode and will use the IP addresses configuration set forth in the MIBs.

See ["Changing a Parameter Value" on page 137](#) for more details.



Note: The recovery mode does not alter any persistent configuration data of the Mediatrix 4100.

Factory Reset

The Factory reset reverts the Mediatrix 4100 back to its default factory settings. It deletes the persistent MIB values of the unit, including:

- ▶ The entire *mediatrixMIBs* configuration.
- ▶ The MIB-II setup.
- ▶ The software download configuration files.
- ▶ The SNMP configuration, including the SNMPv3 passwords and users.

The Factory reset creates a new configuration file with the default factory values. It should be performed with the Mediatrix 4100 connected to a network with access to a DHCP server. If the unit cannot find a DHCP server, it sends requests indefinitely.

You can disable the Factory reset to avoid users deleting the existing configuration. See ["Disabling the Factory Reset" on page 22](#) for more details.

▶ To trigger the Factory Reset:

1. Insert a small, unbent paper clip into the *Reset / Default* switch hole of the Mediatrix 4100. Do not depress before all the LEDs stop blinking and are steadily ON.

2. Release the paper clip.

The Mediatrix 4100 restarts.

This procedure resets all variables in the MIB modules to their default value; defaults include the *localhostSelectConfigSource* variable set to **dhcp**.

When the Mediatrix 4100 has finished its provisioning sequence, it is ready to be used with a DHCP-provided IP address and MIB parameters.



Note: The Factory reset alters any persistent configuration data of the Mediatrix 4100.

Disabling the Factory Reset

You can disable the factory reset procedure, even if users depress the *Reset / Default* switch. Disabling the factory reset means that users will not be able to revert the Mediatrix 4100 back to its factory settings if there are configuration problems.

► To change the factory reset behaviour:

1. In the *sysAdminMIB*, set the *sysAdminDefaultSettingsEnable* variable to **disable**.
In this case, users can only perform a Recovery Mode procedure. See [“Reset / Default Switch” on page 19](#) for more details.

Software Restart

You can initiate a software restart of the Mediatrix 4100 by using MIB parameters.



In the *Unit Manager Network Administration Manual*, refer to chapter *Performing Actions on Mediatrix Units*, section *Restarting a Unit*.

► To perform a software restart:

1. In the *groupAdminMIB*, locate the *groupAdminMIBObjects* group.
2. Set the *groupSetAdmin* variable to the appropriate type of restart:
 - *Locked*: waits for the state of all lines to be locked, and then restarts. This is called a graceful restart.
 - *ForceLock*: restarts immediately. This is called an abrupt restart.
 - *Unlock*: the command is discarded.
3. Set the *groupReset* variable to **SoftReset**.
The Mediatrix 4100 restarts.

Restart Behaviour

This feature affects the behaviour of the Mediatrix 4100 when it restarts.

You can instruct the Mediatrix 4100 to check its TCP/IP stack before declaring the restart successful.

This could be useful when the unit is subjected to a broadcast storm (such as a TCP/IP flood) while it is restarting. In this case, and when the TCP/IP stack check is enabled, the unit enters into the rescue mode and cannot be contacted through SNMP. You thus need to restart the Mediatrix 4100 manually. However, when the TCP/IP stack check is disabled, a broadcast storm during a restart will cause the unit to continuously restart until the storm subsides.

► To define the restart behaviour:

1. In the *bootBehaviorMIB*, enable the *checkTcpIpStackForSuccessfulBoot* variable.
When the variable is enabled, the TCP/IP stack must initialize properly to consider the restart a success. In a flood scenario, the unit may end up in the rescue mode.
When the feature is disabled, even if the TCP/IP stack fails to initialize during a TCP/IP flood, the restart is considered successful and the unit does not enter into the rescue mode.

Verifying the Installation

There are two ways to verify that the Mediatrix 4100 is properly connected to the IP network and is working:

- ▶ By contacting it with a SNMP Browser
- ▶ By pinging it

These two procedures assume that you know the IP address of the Mediatrix 4100 you want to verify. If the Mediatrix 4100 does not respond, do the following:

- ▶ Verify that the LAN cable is securely connected to the Mediatrix 4100 and to the network connector.
- ▶ Be sure that you did not connect a crossover network cable.
- ▶ Verify the state of the IP network to ensure it is not down (the *LAN* LED should be ON or blinking).

Web Interface – Introduction

The Mediatrix 4100 contains an embedded web server to set parameters by using the HTTP protocol.

Standards Supported

RFC 2616 – Hypertext Transfer protocol - HTTP/1.1.

Introduction

The web interface may be used to:

- ▶ View the status of the Mediatrix 4100.
- ▶ Set numerous parameters of the Mediatrix 4100.

All parameters located in the web interface may also be configured via SNMP. Each section of the web interface suggests a link to the corresponding SNMP section.

Before using the web-based configuration service, you must ensure that it is enabled.

▶ To enable the web-based configuration service:

1. In the *ipAddressConfig* folder, set the TCP port on which to listen for HTTP requests in the *httpServerPort* variable (under the *ipAddressConfigHttpEngine* group).
2. In the *httpServerMIB*, enable the service by setting the *httpServerEnable* variable to **enable**.

Using the Web Interface

Media5 recommends that you use the latest version of the Microsoft[®] Internet Explorer web browser to properly access the web interface.

Web Interface Access Limitation

Access to the web interface can be limited to only one of the Mediatrix 4100's interface or all its interfaces. Furthermore, you can grant access to one or both of the Administration and End User web pages available. This can be modified only by using the proper MIB variable.

▶ To limit the access to the web interface:

1. In the *httpServerMIB*, configure the interface where the web pages can be accessed in the *httpServerAccess* variable.

You have the following choices:

Table 13: Web Access Limitation Parameters

Access	Description
lanOnly	You can access the web interface from the LAN side, which is usually associated with the <i>ETH1</i> connector.
wanOnly	You can access the web interface from the WAN side, which is usually associated with the <i>ETH2</i> connector.
all	You can access the web interface from both the LAN and WAN sides.

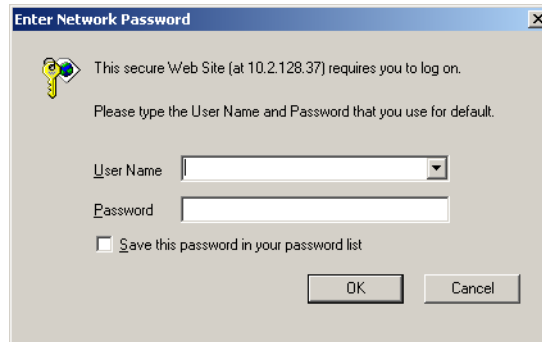
2. Restart the Mediatrix 4100 so that the changes may take effect.

► **To use the web interface configuration:**

1. In your web browser's address field, type the IP address of the Mediatrix 4100 (if you have performed a recovery mode, this is **192.168.0.1**).

The following opens:

Figure 11: Login Window



The Mediatrix 4100 uses Digest Authentication to grant access to the web-based interface.

2. Enter the proper user name and password.

The user name and password must be valid. They are case sensitive hence they must be entered properly. Default factory values are:

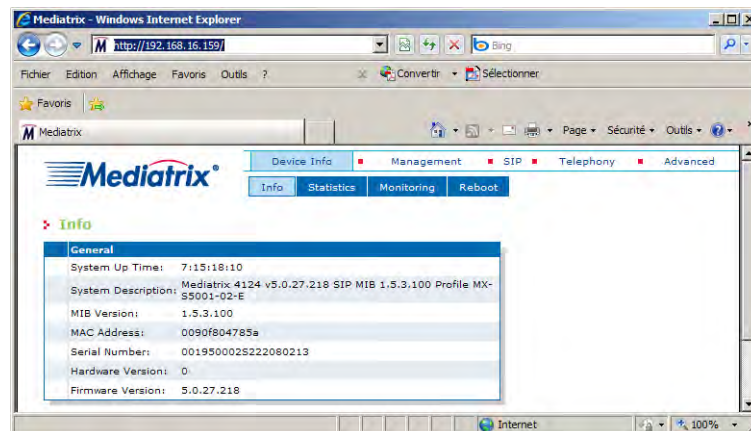
- **User Name:** admin
- **Password:** 1234

Once you have accessed the web interface, you can change the password as described in [“Admin Page” on page 31](#).

3. Click OK.

The *Device Info* web page displays. It has three sub-pages: *Info*, *Statistics*, and *Monitoring*. It stays accessible for as long as the Internet browser used to access the Mediatrix 4100 web interface is opened.

Figure 12: Device Info Web Page



System Status of the Mediatrix 4100

The *Device Info – Info* page displays the current system status of the Mediatrix 4100.

Table 14: Info Page

IP Information	Description
System Up Time	The time (in hundredths of a second) since the network management portion of the system was last re-initialized.
System Description	A textual description of the Mediatrix 4100. It usually includes the full name and version identification of its hardware type, software operating-system, and networking software.
MIB Version	MIB version currently loaded in the Mediatrix 4100.
MAC Address	MAC address of the Mediatrix 4100.
Serial Number	Serial number of the Mediatrix 4100.
Hardware version	Version of the analog circuit board of the Mediatrix 4100.
Firmware version	Software version of the Mediatrix 4100.

Menu Frame

The Menu frame is displayed at the top of the browser window. It contains management links that allow you to display web pages in the Content frame.

Table 15: Menu Frame Links

Link	Description
Device Info	Info: Links to the <i>Info</i> sub-page of the <i>Device Info</i> web page, which displays, in read-only format, the system parameters of the Mediatrix 4100.
	Statistics: Links to the <i>Statistics</i> sub-page of the <i>Device Info</i> web page, which displays, in read-only format, various communication statistics of the Mediatrix 4100.
	Monitoring: Links to the <i>Monitoring</i> sub-page of the <i>Device Info</i> web page, which allows you to configure the syslog parameters of the Mediatrix 4100. See “Syslog Monitoring” on page 29 for more details.
Management	Admin: Links to the <i>Admin</i> sub-page of the <i>Management</i> web page, which allows you to configure line administration parameters of the Mediatrix 4100. See “Admin Page” on page 31 for more details.
	Network Settings: Links to the <i>Network Settings</i> sub-page of the <i>Management</i> web page, which allows you to configure network-related parameters of the Mediatrix 4100, such as IP address source, IP address, etc. See “Network Settings” on page 35 for more details.
	Configuration File: Links to the <i>Configuration File</i> sub-page of the <i>Management</i> web page, which allows you to configure the various configuration file download parameters of the Mediatrix 4100. See “Configuration File Download” on page 40 for more details.
	Firmware Download: Links to the <i>Firmware Download</i> sub-page of the <i>Management</i> web page, which allows you to configure the various firmware download parameters of the Mediatrix 4100. See “Firmware Download” on page 51 for more details.

Table 15: Menu Frame Links (Continued)

Link	Description
SIP	Configuration: Links to the <i>Configuration</i> sub-page of the <i>SIP</i> web page, which allows you to configure the SIP server and SIP user agent parameters of the Mediatrix 4100. See “SIP Servers Configuration” on page 63 for more details.
	Interop: Links to the <i>Interop</i> sub-page of the <i>SIP</i> web page, which allows you to configure the various interoperability features of the Mediatrix 4100. See “SIP Interop” on page 67 for more details.
	Authentication: Links to the <i>Authentication</i> sub-page of the <i>SIP</i> web page, which allows you to configure authentication parameters of the Mediatrix 4100. See “SIP Authentication” on page 71 for more details.
Telephony	Digit Maps: Links to the <i>Digit Maps</i> sub-page of the <i>Telephony</i> web page, which allows you to configure the various digit maps of the Mediatrix 4100. See “Digit Maps” on page 73 for more details.
	CODEC: Links to the <i>CODEC</i> sub-page of the <i>Telephony</i> web page, which allows you to configure the voice and data codec-related parameters of the Mediatrix 4100. See “Voice & Fax Codecs” on page 79 for more details.
	Call Forward: Links to the <i>Call Forward</i> sub-page of the <i>Telephony</i> web page, which allows you to configure the call forward on busy, on no answer, and unconditional parameters of the Mediatrix 4100. See “Call Forward” on page 90 for more details.
	Services: Links to the <i>Services</i> sub-page of the <i>Telephony</i> web page, which allows you to configure the subscriber services of the Mediatrix 4100. See “Services” on page 96 for more details.
	Misc: Links to the <i>Misc</i> sub-page of the <i>Telephony</i> web page, which allows you to configure advanced telephony attributes of the Mediatrix 4100. See “Miscellaneous” on page 104 for more details.
Advanced	QoS: Links to the <i>QoS</i> sub-page of the <i>Advanced</i> web page, which allows you to configure the Quality of Service parameters of the Mediatrix 4100. See “Quality of Service (QoS)” on page 117 for more details.
	Emergency: Links to the <i>Emergency</i> sub-page of the <i>Advanced</i> web page, which allows you to configure the Emergency Call parameters of the Mediatrix 4100. See “Emergency Page” on page 120 for more details.
	STUN: Links to the <i>STUN</i> sub-page of the <i>Advanced</i> page, which allows you to configure the STUN client of the Mediatrix 4100. See “STUN Configuration” on page 121 for more details.

Content Frame

The Content frame is displayed in the lower part of the browser window. It contains the various web pages that allow you to manage the Mediatrix 4100.

Submitting Changes

When you perform changes in the web interface and click the *Submit* button, the Mediatrix 4100 validates the changes. A message informs you that the Mediatrix 4100 must be restarted if one or more non-dynamic value was changed. If at least one modified value is invalid, a message informs you that some values are invalid. Most changes are not dynamic and require to restart the Mediatrix 4100.

The Reboot page then opens. You must click *Reboot*.

This restarts the Mediatrix 4100. If the unit is in use when you click *Reboot*, all calls are terminated.

Syslog Monitoring

Standards Supported

RFC 3164 – The BSD Syslog Protocol

The *Monitoring* sub-page of the *Device Info* page allows you to set the Syslog daemon configuration of the Mediatrix 4100. You can also set these parameters and supplemental information via SNMP, as described in [“Chapter 28 - Syslog Daemon” on page 341](#).

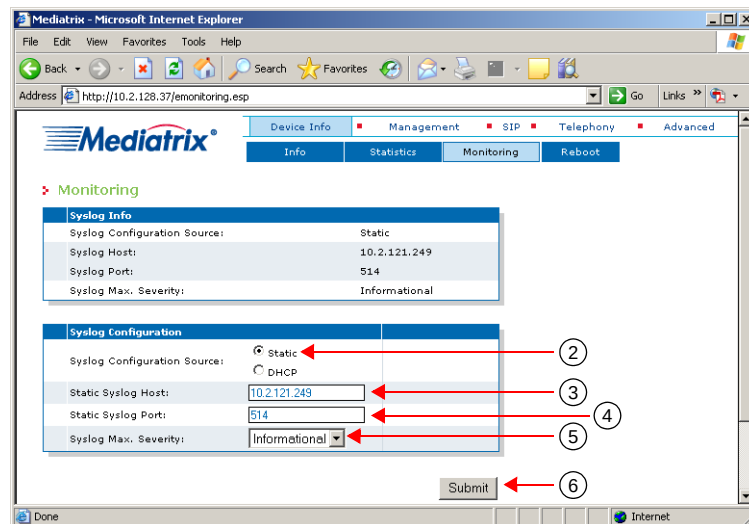
The Syslog daemon is a general purpose utility for monitoring applications and network devices with the TCP/IP protocol. With this software, you can monitor useful messages coming from the Mediatrix 4100 unit. If no Syslog daemon address is provided by a DHCP server or specified by the administrator, no messages are sent.

For instance, if you want to download a new software into the Mediatrix 4100, you can monitor each step of the software download phase. Furthermore, if the unit encounters an abnormal behaviour, you may see accurate messages that will help you troubleshoot the problem.

► To set the syslog parameters:

1. In the web interface, click the *Device Info* link, then the *Monitoring* sub-link.

Figure 13: Device Info – Monitoring Web Page



2. Select the configuration source of the syslog information in the *Syslog Configuration Source* choices.

Table 16: Syslog Configuration Sources

Source	Description
Static	You manually enter the value and it remains the same every time the Mediatrix 4100 restarts. Use the static configuration if you are not using a DHCP server or if you want to bypass it.
DHCP	The value is obtained at start-time by querying a DHCP server and using standard DHCP fields or options. Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See “Chapter 8 - IP Address and Network Configuration” on page 143 for more details.

3. If the Syslog configuration source is **Static**, enter the Syslog server static IP address or domain name in the *Static Syslog Host* field.
4. If the Syslog configuration source is **Static**, enter the Syslog server static IP port number in the *Static Syslog Port* field.
5. Set the syslog severity level in the *Syslog Max. Severity* choices.
This indicates which syslog message is processed. Any syslog message with a severity value greater than the selected value is ignored by the agent.
 - Disabled
 - Critical
 - Error
 - Warning
 - Informational
 - Debug

A higher level mask includes lower level masks, e.g., *Warning* includes *Error* and *Critical*.
6. Click *Submit* if you do not need to set other parameters.



Note: The current syslog information is displayed in the *Syslog Info* section.

The following are some of the messages the unit sends:

Table 17: Syslog Messages Examples

Event	Level	Message
The configuration update with the specific configuration file has been successful (configuration file fetching)	Informational	The specific configuration update succeeded.
The configuration update with the specific configuration file experienced an error and has not been completed (configuration file fetching)	Error	The specific configuration update failed.
The software update has been successful	Informational	The software update succeeded.
The software update experienced an error and has not been completed	Error	The software update failed.

Configuring the Syslog Daemon Application

You shall configure the Syslog daemon to capture those messages. Refer to your Syslog daemon's documentation to learn how to properly configure it to capture messages.

Web Interface – Management

The *Management* section of the web interface allows you to configure general parameters of the Mediatrix 4100, as well as its configuration file download and firmware download parameters.

Admin Page

The *Admin* sub-page of the *Management* page allows you to configure line administration parameters of the Mediatrix 4100 grouped in four categories:

- ▶ HTTP Server Username/Password
- ▶ System Management
- ▶ Group Port Management
- ▶ Interface Management

HTTP Server Password

Standards Supported

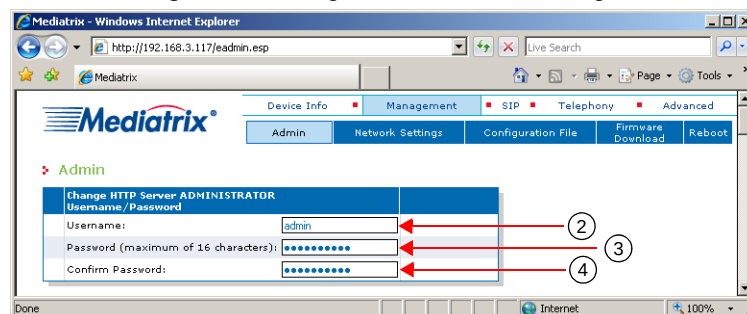
RFC 2617 – HTTP Authentication: Basic and Digest Access Authentication

The *Change HTTP Server ADMINISTRATOR Username/Password* section allows you to modify the default password to access the web interface. The Mediatrix 4100 supports basic HTTP authentication, as described in RFC 2617.

▶ To change the username/password:

1. In the web interface, click the *Management* link, then the *Admin* sub-link.

Figure 14: Management – Admin Web Page



2. Enter the new user name.
3. Enter the new password.
The password cannot begin with "0" or exclusively be made up of several "0".
The password is case sensitive. It can be a string of 0 to 16 characters. All characters are allowed. However, some special characters, such as accented characters (é, à, etc.), may not work.
4. Retype the password in the *Confirm Password* field.

5. Click *Submit* if you do not need to set other parameters.

The password resets back to the default value when:

- Resetting the password by using the *httpServerResetToDefaultPwd* variable (see [“Default User Name and Password” on page 32](#) for more details).
- Performing a factory reset (see [“Factory Reset” on page 21](#) for more details).

Default User Name and Password

The default user name and password the web interface uses are stored in MIB variables you can modify.

► To modify the default user name and password:

1. In the *httpServerMIB*, set the default user name for the web interface access authentication in the *httpServerUsername* variable.
2. Set the default password for the web interface access authentication in the *httpServerDefaultPassword* variable.
Both changes are immediate and take effect on all new web accesses.

► To reset the web authentication password to the default value:

1. In the *httpServerMIB*, set the *httpServerResetToDefaultPwd* variable to **reset**.
The web password is reset to the default value specified by the *httpServerDefaultPassword* variable.
2. Restart the Mediatrix 4100 so that the change may take effect.

Issue: Factory Reset does not Reset the Default Password Value

The following describes three cases in which the factory reset may not properly reset the HTTP server password. Each case defines the password you must use to access the web interface.

Case #1

You do not modify the password via the Web page and you upgrade to a new software version with a new default password in the profile.

Table 18: Case 1 Issue

Item	Description
Wanted Behaviour	The password to use is the default password in the new profile.
Current Behaviour	The password to use is the default password in the <i>previous</i> version of the profile.
Workaround	Once using the new software version, use the <i>httpServerResetToDefaultPwd</i> variable as described in “Default User Name and Password” on page 32 .

Case #2

You modify the password via the web interface and you upgrade to a new software version with a new default password in the profile.

Table 19: Case 2 Issue

Item	Description
Wanted Behaviour	The password to use is the password modified via the web interface in the older software version.
Current Behaviour	Same as the wanted behaviour.
Workaround	None required.

Case #3

You are performing a factory reset.

Table 20: Case 3 Issue

Item	Description
Wanted Behaviour	The password to use is the default password in the profile.
Current Behaviour	Same as the wanted behaviour.
Workaround	Once using the new software version, use the <code>httpServerResetToDefaultPwd</code> variable as described in “Default User Name and Password” on page 32 .

System Management

The following are the system management parameters you can set. These parameters apply to the whole Mediatrix 4100.

You can also set these parameters via SNMP, as described in [“Chapter 7 - MIB Structure and SNMP” on page 127](#).

► **To set the system management parameters:**

1. In the *System Management* section of the *Admin* page, define the SNMP service agent status in the *SNMP Agent Activation* choices.

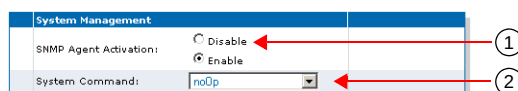
The SNMP service agent controls whether the Mediatrix 4100 may be accessed via SNMP or not.

- Enable: SNMP connections are allowed.
- Disable: SNMP connections are forbidden.

If you are exclusively using the web page to configure the Mediatrix 4100, you could disable the SNMP service agent and forbid any SNMP connection to the unit.

If the SNMP agent is disabled and the unit enters into recovery mode, SNMP connections are allowed during that time. Once the unit returns to its normal state, the SNMP service agent will revert to the configuration you have set.

Figure 15: Management – System Management Web Page



2. Select the proper command to execute in the *System Command* drop-down menu. This command controls the various commands that can be performed by the unit.

Table 21: System Commands

Command	Description
noOp	No action is taken.
checkRam	Launches the RAM check routine.
checkRom	Currently unused.
downloadSoftware	Launches a firmware update. See “Firmware Download” on page 51 for more details.
resetStats	Resets all cumulated call statistics.
setConfigSourcesStatic	Sets all configuration sources supported by the unit to “static”. This command can be used when no DHCP server is present in the network to easily configure the unit to use static values.

Table 21: System Commands (Continued)

Command	Description
updateConfiguration	Downloads configuration files. See "Configuration File Download" on page 40 for more details.

- Click *Submit* if you do not need to set other parameters.

Group Port Management

You can set the administrative state of all the lines of the Mediatix 4100.

► To set the group port management parameters:

- In the *Group Port Management* section of the *Admin* page, select the proper command to execute in the *Group Port Command* field.

Figure 16: Group Port Management Section

This command locks/unlocks all the lines of the Mediatix 4100. This state is kept until the unit restarts. It offers the following settings:

Table 22: Group Port Settings

Setting	Description
noOp	No action is taken.
unlockAllPorts	Registers all the lines to the SIP server.
lockAllPorts	Cancels all the lines registration to the SIP server. Active calls in progress remain established until normal call termination. No new calls may be initiated.

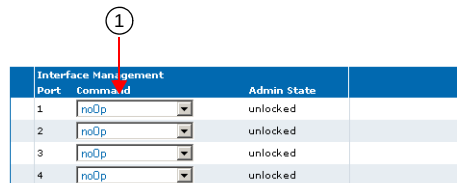
- Click *Submit* if you do not need to set other parameters.

Interface Management

You can set the administrative state of a line that will be kept until the Mediatix 4100 restarts.

► To set the interface management parameters:

- In the *Interface Management* section of the *Admin* page, select the proper command to execute in the *Command* column.

Figure 17: Interface Management Section

This command temporary locks/unlocks the selected line of the Mediatrix 4100. This state is kept until the unit restarts. It offers the following settings:

Table 23: Temporary Lock Settings

Setting	Description
noOp	No action is taken.
unlock	Registers the line to the SIP server.
lock	Cancels the line registration to the SIP server. Active calls in progress remain established until normal call termination. No new calls may be initiated.
forcelock	Cancels the line registration to the SIP server. All active calls in progress are terminated immediately. No new calls may be initiated.

- Click *Submit* if you do not need to set other parameters.

Network Settings

The *Network Settings* sub-page of the *Management* page allows you to configure network-related parameters of the Mediatrix 4100 grouped in three categories:

- ▶ Ethernet
- ▶ Network Settings
- ▶ SNTP

Ethernet Connection Speed

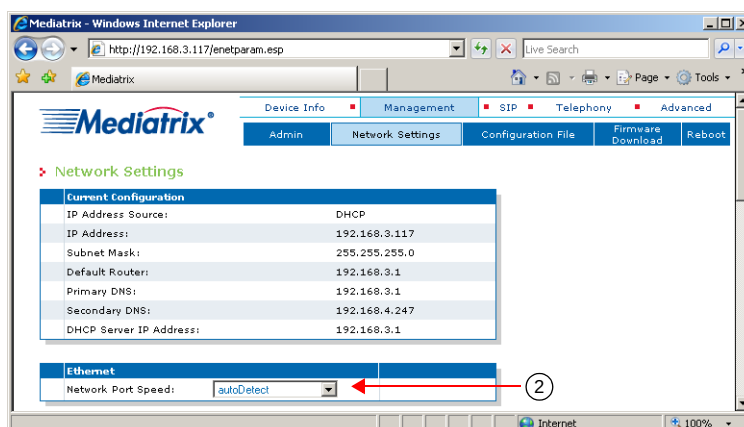
You can set the speed of the Mediatrix 4100's Ethernet connection.

You can also set this parameter via SNMP, as described in ["Ethernet Connection Speed" on page 162](#).

- ▶ **To set the Ethernet connection speed:**

- In the web interface, click the *Management* link, then the *Network Settings* sub-link.

Figure 18: Management – Network Settings Web Page



- Set the Ethernet connection speed of the *Ethernet* connector in the *Network Port Speed* field. The following values are available:
 - Auto detect
 - 10Mbps-HalfDuplex

- 100Mbps-HalfDuplex
- 10Mbps-FullDuplex
- 100Mbps-FullDuplex

A half-duplex connection refers to a transmission using two separate channels for transmission and reception, while a full-duplex connection refers to a transmission using the same channel for both transmission and reception.

If unknown, select **Auto detect** so that the Mediatrix 4100 can automatically detect the network speed.



Caution: Whenever you force a connection speed / duplex mode, be sure that the other device and all other intermediary nodes used in the communication between the two devices have the same configuration. See [“Speed and Duplex Detection Issues” on page 36](#) for more details.

3. Click *Submit* if you do not need to set other parameters.

Speed and Duplex Detection Issues

There are two protocols for detecting the Ethernet link speed:

- ▶ An older protocol called parallel detection.
- ▶ A more recent protocol called auto-negotiation (IEEE 802.3u).

The auto-negotiation protocol allows to detect the connection speed and duplex mode. It exchanges capabilities and establishes the most efficient connection. When both endpoints support the auto-negotiation, there are no problems. However, when only one endpoint supports auto-negotiation, the parallel detection protocol is used. This protocol can only detect the connection speed; the duplex mode cannot be detected. In this case, the connection may not be established.

The Mediatrix 4100 has the possibility to force the desired Ethernet link speed and duplex mode by disabling the auto-negotiation and selecting the proper setting. When forcing a link speed at one end, be sure that the other end (a hub, switch, etc.) has the same configuration. To avoid any problems, the link speed and duplex mode of the other endpoint must be exactly the same.

Network Settings

The *Network Settings* section allows you to set the IP information the Mediatrix 4100 needs to work properly. This section is vital to the proper operation of the Mediatrix 4100. If a field of this group is not properly set, the Mediatrix 4100 may not be able to restart and be contacted after it has restarted.

You can also set this parameter via SNMP, as described in [“Chapter 8 - IP Address and Network Configuration” on page 143](#).

▶ To set network parameters:

1. In the *Network Settings* section of the *Network Settings* page, select the configuration source of the network information in the *IP Address Source* choices.

Table 24: Network Settings Configuration Sources

Source	Description
Static	You manually enter the value and it remains the same every time the Mediatrix 4100 restarts. Use the static configuration if you are not using a DHCP server or if you want to bypass it.
DHCP	The value is obtained at start-time by querying a DHCP server and using standard DHCP fields or options. Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See “Chapter 8 - IP Address and Network Configuration” on page 143 for more details.

Figure 19: Network Settings Section

Network Settings	
IP Address Source:	☐ Static ☒ DHCP
IP Address:	192.168.0.1
Subnet Mask:	255.255.255.0
Default Router:	192.168.0.10
Primary DNS:	192.168.0.10
Secondary DNS:	192.168.0.10
SNMP Port:	161

- If the IP address source is **Static**, enter the following static IP information.

Table 25: IP Addresses Parameters

Parameter	Definition
IP Address	Public IP address of the Mediatrix 4100. This address is used for incoming signalling, media and management traffic.
Subnet Mask	Subnet mask IP address used by the Mediatrix 4100. Note: Media5 recommends not to set a subnet mask of 255.255.255.254 because this would only create a subnet with two addresses. This only leaves one bit host addresses. Since a subnet must have a network (all bits 0) and a broadcast address (all bits 1), this leaves no room for hosts.
Default Router	Default router IP address used by the Mediatrix 4100.
Primary DNS	Primary Domain Name Server IP address used by the Mediatrix 4100.
Secondary DNS	Secondary Domain Name Server IP address used by the Mediatrix 4100.

- Enter the default SNMP agent port in the *SNMP Port* field.
This is the port number to use to reach the local host via the SNMP protocol.
- Click *Submit* if you do not need to set other parameters.



Note: The current configuration file server information is displayed in the *Current Configuration* section.

SNTP Settings

Standards Supported

RFC 1769 – Simple Network Time Protocol (SNTP)

The Simple Network Time Protocol (SNTP) enables the notion of time (date, month, time) into the Mediatrix 4100. It updates the internal clock of the unit, which is the client of a SNTP server. It is required when dealing with features such as the caller ID.

SNTP is used to synchronize a SNTP client with a SNTP or NTP server by using UDP as transport.

You can also set these parameters via SNMP, as described in [“Chapter 20 - SNTP Settings” on page 289](#).

► To set the SNTP client of the Mediatrix 4100:

- In the *SNTP* section of the *Network Settings* page, select **Enable** in the *SNTP Enable* choices.

Figure 20: SNTP Section

SNTP	
SNTP Enable:	☐ Enable ☒ Disable
SNTP Source:	DHCP
SNTP Host:	192.168.0.10
SNTP Port:	123
SNTP Timezone:	EST5DST4M41.0/02:0
SNTP Synchronization Period:	1440
SNTP Synchronization Period on Error:	60

2. Select the configuration source of the SNTP information in the *SNTP Source* choices.

Table 26: Network Settings Configuration Sources

Source	Description
Static	You manually enter the value and it remains the same every time the Mediatrix 4100 restarts. Use the static configuration if you are not using a DHCP server or if you want to bypass it.
DHCP	The value is obtained at start-time by querying a DHCP server and using standard DHCP fields or options. Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See “Chapter 8 - IP Address and Network Configuration” on page 143 for more details.

3. If the SNTP source is **Static**, enter the following static IP information.

Table 27: SNTP Static Address

Field	Description
SNTP Host	Static SNTP server IP address or domain name.
SNTP Port	Static SNTP server IP port number.

4. Enter a valid string in the *SNTP Timezone* field.
 The format of the string is validated upon entry. Invalid entries are refused. The default value is:
`EST5DST4,M4.1.0/02:00:00,M10.5.0/02:00:00`
 A POSIX string is a set of standard operating system interfaces based on the UNIX operating system. The format of the IEEE 1003.1 POSIX string is defined in the <bootp-dhcp-option-88.txt> Internet draft as:
`STD0OFFSET[DST[OFFSET], [START[/TIME],END[/TIME]]]`
 Refer to the following sub-sections for explanations on each part of the string.
5. Set the synchronization information:

Table 28: SNTP Synchronization Information

Field	Description
SNTP Synchronisation Period	Time interval (in minutes) between requests made to the SNTP server. The result is used to synchronize the unit with the time server.
SNTP Synchronisation Period on Error	Time interval (in minutes) between retries after an unsuccessful attempt to reach the SNTP server.

6. Click *Submit* if you do not need to set other parameters.
 The current configuration file server information is displayed in the *SNTP Info* section.

STD / DST

Three or more characters for the standard (STD) or alternative daylight saving time (DST) time zone. Only STD is mandatory. If DST is not supplied, the daylight saving time does not apply. Lower and upper case letters are allowed. All characters are allowed except digits, leading colon (:), comma (,), minus (-), plus (+), and ASCII NUL.

OFFSET

Difference between the GMT time and the local time. The offset has the format *h[h][:m[m][:s[s]]]*. If no offset is supplied for DST, the alternative time is assumed to be one hour ahead of standard time. One or more digits can be used; the value is always interpreted as a decimal number.

The hour value must be between 0 and 24. The minutes and seconds values, if present, must be between 0 and 59. If preceded by a minus sign (-), the time zone is east of the prime meridian, otherwise it is west, which can be indicated by the preceding plus sign (+). For example, New York time is GMT 5.

START / END

Indicates when to change to and return from the daylight saving time. The *START* argument is the date when the change from the standard to the daylight save time occurs; *END* is the date for changing back. If *START* and *END* are not specified, the default is the US Daylight saving time start and end dates. The format for start and end must be **one** of the following:

- ▶ **n** where *n* is the number of days since the start of the year from 0 to 365. It must contain the leap year day if the current year is a leap year. With this format, you are responsible to determine all the leap year details.
- ▶ **Jn** where *n* is the Julian day number of the year from 1 to 365. Leap days are not counted. That is, in all years – including leap years – February 28 is day 59 and March 1 is day 60. It is impossible to refer to the occasional February 29 explicitly. The *TIME* parameter has the same format as *OFFSET* but there can be no leading minus (-) or plus (+) sign. If *TIME* is not specified, the default is 02:00:00.
- ▶ **Mx[x].y.z** where *x* is the month, *y* is a week count (in which the *z* day exists) and *z* is the day of the week starting at 0 (Sunday). As an example:

M10.4.0

is the fourth Sunday of October. It does not matter if the Sunday is in the 4th or 5th week.

M10.5.0

is the last Sunday of October (5 indicates the last *z* day). It does not matter if the Sunday is in the 4th or 5th week.

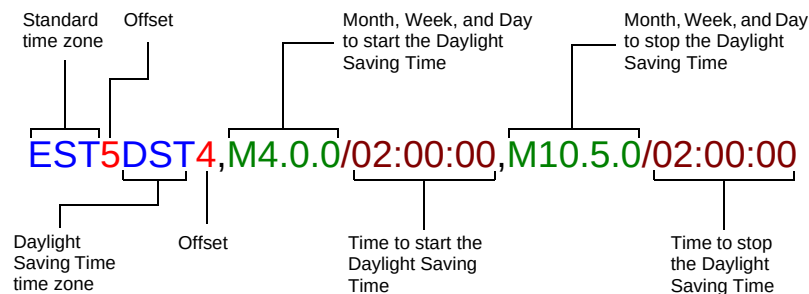
M10.1.6

is the first week with a Saturday (thus the first Saturday). It does not matter if the Saturday is in the first or second week.

The *TIME* parameter has the same format as *OFFSET* but there can be no leading minus (-) or plus (+) sign. If *TIME* is not specified, the default is 02:00:00.

Example

The following is an example of a proper POSIX string:



The following are some valid POSIX strings:

Table 29: Valid POSIX Strings

Time Zone	POSIX String
Pacific Time (Canada & US)	PST8PDT7,M3.2.0/02:00:00,M11.1.0/02:00:00
Mountain Time (Canada & US)	MST7MDT6,M3.2.0/02:00:00,M11.1.0/02:00:00
Central Time (Canada & US)	CST6CDT5,M3.2.0/02:00:00,M11.1.0/02:00:00
Eastern Time Canada & US)	EST5EDT4,M3.2.0/02:00:00,M11.1.0/02:00:00

Table 29: Valid POSIX Strings (Continued)

Time Zone	POSIX String
Atlantic Time (Canada)	AST4ADT3,M3.2.0/02:00:00,M11.1.0/02:00:00
GMT Standard Time	GMT0DMT-1,M3.5.0/01:00:00,M10.5.0/02:00:00
W. Europe Standard Time	WEST-1DWEST-2,M3.5.0/02:00:00,M10.5.0/03:00:00
China Standard Time	CST-8
Tokyo Standard Time	TST-9
Central Australia Standard Time	CAUST-9:30DCAUST-10:30,M10.5.0/02:00:00,M3.5.0/02:00:00
Australia Eastern Standard Time	AUSEST-10AUSDST-11,M10.5.0/02:00:00,M3.5.0/02:00:00
UTC (Coordinated Universal Time)	UTC0

Configuration File Download

The configuration file download feature allows to update the Mediatrix 4100 configuration by transferring a configuration file via TFTP or HTTP. The configuration file is transferred from the configuration file download server and the Mediatrix 4100 is the session initiator. The advantage of having the Mediatrix 4100 as the session initiator is to allow NAT traversal.

The *Configuration File* sub-page of the *Management* page allows you to set various configuration file download parameters grouped in three categories:

- ▶ General parameters
- ▶ Encryption
- ▶ Automatic Update

You can also set these parameters via SNMP, as described in [“Chapter 12 - Configuration File Download” on page 187](#).

Configuration File Download Server

The service allows to download a unique file for each Mediatrix 4100, and/or a file shared among many units. These configuration files may be encrypted or not.

You have the choice to perform the configuration file download by using the TFTP protocol or the HTTP protocol. You can also configure the Mediatrix 4100 to automatically update its configuration.

To download a configuration file, you may need to setup the following applications on your computer:

- ▶ TFTP server with proper root path
- ▶ SNTP server properly configured
- ▶ HTTP server with proper root path

Configuring the TFTP Server

If you are to perform a configuration file download by using the TFTP protocol, you must install a TFTP (Trivial File Transfer Protocol) server running on the PC designated as the TFTP server host. It is assumed that you know how to set the TFTP root path. If not, refer to your TFTP server's documentation.

Configuring the SNTP Server

If you are to use the automatic configuration file update feature (see [“Automatic Configuration Update” on page 46](#) for more details), you need to have a time server SNTP that is accessible and properly configured. It is assumed that you know how to configure your SNTP server. If not, refer to your SNTP server's documentation. You can also refer to [“SNTP Settings” on page 37](#) for more details on how to configure the Mediatrix 4100 for a SNTP server.

Configuring the HTTP Server

If you are to perform a configuration file download by using the HTTP protocol, you must install a HTTP server running on the PC designated as the server host. It is assumed that you know how to set the root path. If not, refer to your HTTP server's documentation.

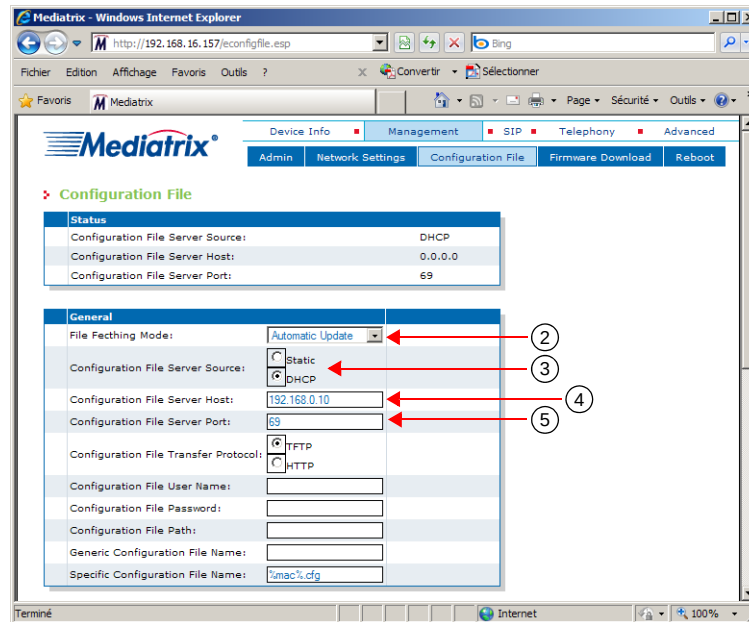
Configuration File Server Settings

The Mediatrix 4100 must know the IP address and port number of its configuration file server. This server contains the configuration file the Mediatrix 4100 will download. You can assign these information to the Mediatrix 4100 through a DHCP server or manually enter them yourself.

► To set the configuration file server parameters:

1. In the web interface, click the *Management* link, then the *Configuration File* sub-link.

Figure 21: Management – Configuration File Web Page



2. Select how configuration files are downloaded in the *File Fetching Mode* drop-down menu.

Table 30: File Fetching Mode Parameters

Parameter	Description
Disabled	Does not perform a configuration file download.
Use Management Server	Configuration files are sent by the management server upon request. The management server is the initiator of the TFTP session.
Automatic Update	Configuration files are automatically fetched by the unit. Refer to "Automatic Configuration Update" on page 46 for details. The unit is the initiator of the transfer sessions. This method facilitates the NAT traversal.

3. Select the configuration source of the configuration download in the *Configuration File Server Source* choices.

Table 31: Configuration File Information Sources

Source	Description
Static	You manually enter the value and it remains the same every time the Mediatrix 4100 restarts. Use the static configuration if you are not using a DHCP server or if you want to bypass it.
DHCP	The value is obtained at start-time by querying a DHCP server and using standard DHCP fields or options. Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See “Chapter 8 - IP Address and Network Configuration” on page 143 for more details.

4. If the configuration file server configuration source is **Static**, enter the configuration file server static IP address or domain name in the *Configuration File Server Host* field.
This is the current address of the PC that hosts the configuration files.
5. If the configuration file server configuration source is **Static**, enter the configuration file server static IP port number in the *Configuration File Server Port* field.
The default port value complies to RFC 1340 on the well-known ports (assigned numbers). This value applies to a TFTP server. It may be different for other servers. If you are using an HTTP server to perform the configuration file download, you must change the port value to 80.
6. Proceed to [“Setting up the Configuration File Download” on page 42](#).

Setting up the Configuration File Download

When performing a configuration file download, you can download two different files:

- ▶ A generic configuration file that should be used to update a large number of units with the same configuration.
- ▶ A specific configuration file that contains the configuration for a single unit, for instance the telephone numbers of its lines.

When both the generic and specific configuration files are downloaded, settings from the specific configuration file always override the settings from the generic configuration file. These files must be located in the same directory.

▶ To setup the configuration file download:

1. In the *General* section of the *Configuration File* page, set the transfer protocol to use in the *Configuration File Transfer Protocol* field.

Figure 22: General Section

General	
File Fetching Mode:	Automatic Update
Configuration File Server Source:	☐ Static ☒ DHCP
Configuration File Server Host:	192.168.16.4
Configuration File Server Port:	89
Configuration File Transfer Protocol:	☒ TFTP ☐ HTTP
Configuration File User Name:	
Configuration File Password:	
Configuration File Path:	
Generic Configuration File Name:	
Specific Configuration File Name:	tmac%.cfg

You have the choice between **TFTP** and **HTTP**.

Your HTTP server may activate some caching mechanism for the file download. This mechanism caches the initial file download for later processing, thus preventing changes or update of the original file by the user. This can cause strange problems if a user wants to edit a configuration file to modify values and upload it immediately. The result will still return the original file and not the new one.

2. If your HTTP server requires authentication to download the configuration file, set the following:
 - The user name in the *Configuration File User Name* field.
 - The password in the *Configuration File Password* field.

The Mediatrix 4100 supports basic and digest HTTP authentication, as described in RFC 2617.

3. Set the path, on the remote server, of the directory where the configuration files are located in the *Configuration File Path* field.

The path is case sensitive hence it must be entered properly.

The path is relative to the root path of the transfer server (*configFileFetchingHost*). Use the "/" character when defining the path to indicate sub-directories.

Let's consider the following example:

- The directory that contains the configuration file is called: **Config_File**.
- This directory is under **C:/Root/Download**.

Table 32: Path Configurations Example

Root Path	Corresponding Path Name
c:/root/download	Config_File
c:/	root/download/Config_File
c:/root	download/Config_File

The following are some tips to help your download process:

- Use the "/" character when defining the path to indicate sub-directories. For instance, *root/download*.
- If you are using the TFTP protocol to download the software, note that some TFTP servers on Windows do not recognize the "/" character and produce an error. In this case, use the "\" character.
- Use basic directory names, without spaces or special characters such as "~", "@", etc., which may cause problems.
- Cut and paste the path and/or name of the directory that contains the extracted files into the configuration file path of the Mediatrix 4100 (you may have to convert "\" into "/" to eliminate typographical errors).

Note that you can define the **C:/Root/Download** part as you want. The file names may also differ from the example shown above.

4. Set the name of the generic configuration file to download in the *Generic Configuration File Name* field.

The file name is case sensitive hence it must be entered properly.

This file should be used to update a large number of units with the same configuration.

If you leave the field empty, the Mediatrix 4100 does not download the generic configuration file.

5. Set the name of the specific configuration file to download in the *Specific Configuration File Name* field.

The file name is case sensitive hence it must be entered properly.

This file should be used to update the configuration of a single unit.

This field may contain macros that are substituted by actual values when downloading the configuration file. Supported macros are:

- %mac%: the MAC address of the unit
- %product%: the product name of the unit

- %%: the character “%”

For instance:

- The “%mac%.xml” value for a Mediatrix 4100 with MAC address “0090F12345AB” will be “0090F12345AB.xml”.
- The value “Hello%%Hi” will result in “Hello%Hi”.
- The value “%%mac%%mac%.xml” will result in “%0090F12345AB%mac%.xml”.

From left to right: the first macro encountered is first substituted, the second macro encountered is then substituted, etc.

When the character “%” is not part of a macro, it is not replaced. The following are examples:

- The value “%mac.xml” stays “%mac.xml”
- The value “Hello%Hi” stays “Hello%Hi”
- The value “%moc%.xml” stays “%moc%.xml”

If the field is empty (after macro substitution), the Mediatrix 4100 does not download the specific configuration file.

6. Click *Submit* if you do not need to set other parameters.

Configuration Update Status

If valid configuration files are successfully downloaded, then the Mediatrix 4100 automatically restarts to apply all the new settings. If the Mediatrix 4100 does not restart, this could mean the download failed or that the configuration in the file is the same as the configuration in the unit.

A lot of information is transmitted as system log (syslog) messages. The following are some of the syslog messages sent by the unit:

Table 33: Configuration File Download Syslog Messages

Level	Message	Event
Informational	The specific configuration update succeeded.	The configuration update with the specific configuration file has been successful.
Error	The specific configuration update failed.	The configuration update with the specific configuration file experienced an error and has not been completed.
Informational	The configuration file “XXX” was successfully fetched.	A configuration file was successfully fetched.
Informational	The unit configuration is not updated. The parameter values defined in the fetched configuration files are identical to the actual unit configuration.	The parameter values defined in the fetched configuration files are identical to the actual unit configuration.
Informational	The generic file “%s” parameter values are not applied. They are either identical to the unit configuration or overwritten by the specific file.	The generic configuration file parameter values are either identical to the unit configuration or overwritten by the specific configuration file.
Warning	None of the parameter values defined in the configuration file “%s” was successfully applied.	No parameter value from a fetched configuration file was successfully applied (e.g., because of bad OIDs).
Informational	Parameter values defined in the configuration file “%s” were successfully applied.	A fetched configuration file was successfully applied.
Informational	The unit is restarting to complete the configuration update.	All necessary fetched configuration files were successfully applied.

Configuration Files Encryption

You can secure the exchange of configuration files between the server and the Mediatrix 4100. A privacy key allows the unit to decrypt a previously encrypted configuration file.

To encrypt a configuration file (generic or specific), you must use the MxCryptFile application. MxCryptFile is a command line tool that encrypts files before sending them to the Mediatrix 4100 unit. Contact your sales representative for more details.

The following describes how to decrypt a previously encrypted generic or specific configuration file. You must have one key for the generic configuration file and another key for the specific configuration file.

► To decrypt a configuration file:

1. In the *Encryption* section of the *Configuration File* page, select **Enable** in the *Configuration File Encryption* field.

Figure 23: Management – Encryption section

The screenshot shows a web interface titled 'Encryption'. It contains three rows of configuration fields. The first row, 'Configuration File Encryption', has two radio buttons: 'Enable' and 'Disable'. A red arrow points to the 'Enable' button, which is labeled with a circled '1'. The second row, 'Generic Configuration File Password', has a text input field with a red arrow pointing to it, labeled with a circled '2'. The third row, 'Specific Configuration File Password', also has a text input field.

The Mediatrix 4100 will be able to decrypt the next encrypted generic or specific configuration file. If you select **Disable**, the configuration file is not decrypted by the unit and the configuration update fails.

2. Set the proper decryption password field with the password used to decrypt the configuration file.

Table 34: Decryption Passwords

Configuration File	Field
Generic	Generic Configuration File Password
Specific	Specific Configuration File Password

The password is encoded in hexadecimal notation. You can thus use characters in the range 0-9, A-F. All other characters are not supported.

Each character encodes 4 bits of the key. For instance, a 32-bit key requires 8 characters.

- If you enter too many bits, the key is truncated to the first 448 bits.
- If you do not enter enough bits, the key is padded with zeros.

For instance, a 32-bit key could look like the following: A36CB299.

This key must match the key used for the encryption of the relevant configuration file.

If the field is empty, the configuration file is not decrypted.

3. Click *Submit* if you do not need to set other parameters.

Configuration Download Procedure

The following steps explain how to download configuration files from the web interface.



Note: The configuration download via TFTP can only traverse NATs of types “Full Cone” or “Restricted Cone”. If the NAT you are using is of type “Port Restricted Cone” or “Symmetric”, the file transfer will not work.

► To download configuration files:

1. Set the configuration file server host and port as defined in [“Configuration File Server Settings” on page 41](#).



Caution: When downloading via HTTP, the configuration file server’s port must be 80. You can see the actual port assigned in the *Status* section of the *Configuration File* page.

If you are using a DHCP server and it did not provide the proper port, reconfigure it with the proper port or use a static configuration. See [“Configuration File Server Settings” on page 41](#) for more details.

2. Place the configuration files to download on the computer hosting the TFTP or HTTP server. These files must be in a directory under the TFTP root path.
3. If not already done, set the transport protocol and configuration file path as described in [“Setting up the Configuration File Download” on page 42](#).
4. Initiate the configuration file download by setting the *System Command* drop-down menu of the System – Admin Web Page to **updateConfiguration**.
The Mediatrix 4100 immediately downloads the configuration files. See [“System Management” on page 33](#) for more details on the system commands.
5. Click *Submit*.

Automatic Configuration Update

You can configure the Mediatrix 4100 to automatically update its configuration. This update can be done:

- Every time the Mediatrix 4100 restarts.
- At a specific time interval you can define.

NAT Variations

NAT treatment of UDP varies among implementations. The four treatments are:

- Full Cone: All requests from the same internal IP address and port are mapped to the same external IP address and port. Furthermore, any external host can send a packet to the internal host by sending a packet to the mapped external address.
- Restricted Cone: All requests from the same internal IP address and port are mapped to the same external IP address and port. Unlike a full cone NAT, an external host (with IP address X) can send a packet to the internal host only if the internal host had previously sent a packet to IP address X.
- Port Restricted Cone: Similar to a restricted cone NAT, but the restriction includes port numbers. Specifically, an external host can send a packet, with source IP address X and source port P, to the internal host only if the internal host had previously sent a packet to IP address X and port P.
- Symmetric: All requests from the same internal IP address and port, to a specific destination IP address and port, are mapped to the same external IP address and port. If the same host sends a packet with the same source address and port, but to a different destination, a different mapping is used. Furthermore, only the external host that receives a packet can send a UDP packet back to the internal host.

For more details on NAT treatments, refer to RFC 3489.

Automatic Update on Restart

The Mediatrix 4100 may download new configuration files each time it restarts.

► **To set the automatic update every time the Mediatrix 4100 restarts:**

1. Set the configuration file server host and port as defined in ["Configuration File Server Settings" on page 41.](#)



Caution: When downloading via HTTP, the configuration file server's port must be 80. You can see the actual port assigned in the *Status* section of the *Configuration File* page.

If you are using a DHCP server and it did not provide the proper port, reconfigure it with the proper port or use a static configuration. See ["Configuration File Server Settings" on page 41](#) for more details.

2. Place the configuration files to download on the computer hosting the HTTP or TFTP server. These files must be in a directory under the root path.
3. If not already done, set the transport protocol and configuration file path as described in ["Setting up the Configuration File Download" on page 42.](#)
4. In the *Automatic Update* section of the *Configuration File* page, select **Enable** in the *Configuration File Update On Restart* field.

Figure 24: Management – Automatic Update section

The automatic configuration update will be performed each time the Mediatrix 4100 restarts.

The unit configuration is only updated if at least one parameter value defined in the downloaded configuration files is different from the actual unit configuration.

5. Click *Submit* if you do not need to set other parameters.

Automatic Update at a Specific Time Interval

You can configure the Mediatrix 4100 to download new configuration files at a specific day and/or time.

► **To set the automatic update at a specific time interval:**

1. Set the configuration file server host and port as defined in ["Configuration File Server Settings" on page 41.](#)



Caution: When downloading via HTTP, the configuration file server's port must be 80. You can see the actual port assigned in the *Status* section of the *Configuration File* page.

If you are using a DHCP server and it did not provide the proper port, reconfigure it with the proper port or use a static configuration. See ["Configuration File Server Settings" on page 41](#) for more details.

2. Place the configuration files to download on the computer hosting the HTTP or TFTP server. These files must be in a directory under the root path.
3. If not already done, set the transport protocol and configuration file path as described in ["Setting up the Configuration File Download" on page 42.](#)
4. In the *Automatic Update* section of the *Configuration File* page, select **Enable** in the *Configuration File Periodic Update* field.

Figure 25: Management – Automatic Update section

5. Set the waiting period between each configuration update in the *Periodic Update Period* field. The time unit for the period is specified in the *Periodic Update Time Unit* field (see Step 6). Available values are from 1 to 48.
6. Define the time base for automatic configuration updates in the *Periodic Update Time Unit* field. You have the following choices:

Table 35: Time Unit Parameters

Parameter	Description
Minutes	Updates the unit's configuration every x minutes. You can specify the x value in the <i>Periodic Update Period</i> field (see Step 5).
Hours	Updates the unit's configuration every x hours. You can specify the x value in the <i>Periodic Update Period</i> field (see Step 5).
Days	Updates the unit's configuration every x days. You can specify the x value in the <i>Periodic Update Period</i> field (see Step 5). You can also define the time of day when to perform the update in the <i>Periodic Update Time Range</i> field (see Step 7).

7. If you have selected **days** in Step 6, set the time of the day when to initiate a configuration update in the *Periodic Update Time Range* field.

The time of the day is based on the *SNTP Timezone* field of the *Management - Network Settings* page (see [“SNTP Settings” on page 37](#) for more details).

You must have a time server SNTP that is accessible and properly configured, or the automatic configuration update feature may not work properly. It is assumed that you know how to configure your SNTP server. If not, refer to your SNTP server's documentation. You can also refer to [“SNTP Settings” on page 37](#) for more details on how to configure the Mediatrix 4100 for a SNTP server.

If a time range is specified, the unit will download the configuration files at a random time within the interval specified.

The format should be one of the following:

```
hh[:mm[:ss]]
hh[:mm[:ss]] - hh[:mm[:ss]]
```

Where:

hh: Hours.
mm: Minutes.
ss: Seconds.

The configuration files are downloaded at the first occurrence of this value and thereafter with a period defined by the *Periodic Update Period* field. Let's say for instance the automatic unit configuration update is set with the time of day at 14h00 and the update period at every 2 days.

- If the automatic update is enabled before 14h00, the first update will take place the same day at 14h00, then the second update two days later at the same hour, and so on.

- If the time range is set to '14:00 - 15:00' and the automatic unit configuration update is enabled within those hours, the first update will take place the following day. This means that a range of '00:00:00 - 23:59:59' will always take place the next day.



Note: The *Periodic Update Time of Day* field is deprecated. It is recommended to use the *Periodic Update Time Range* field because it has precedence over this field.

8. Click *Submit* if you do not need to set other parameters.

Error Handling

The following configuration file fetching service error sources are divided in three types depending on the transfer protocol: common errors (Table 35), TFTP errors (Table 36) and HTTP errors (Table 37). The error cause and the unit behaviour are also described.

Table 36: Configuration File Fetching Error Handling

Error Type	Cause	Behaviour
Common Error Handling		
Invalid file format	The file format is not valid.	Send a syslog warning message including the file location/name with the transfer server address: The fetched configuration file "XXX", from server "XXX", has an invalid format. No recorded settings applied.
Empty file	Committing an empty file.	Send a syslog warning message including the file location/name with the transfer server address: The fetched configuration file "XXX", from server "XXX", is empty.
Invalid file content	The file contains invalid characters. Allowed characters are ASCII codes 10 (LF), 13(CR), and 32 to 126.	Send a syslog warning message including the file location/name, the transfer server address and the invalid character (ASCII code): The fetched configuration file "XXX", from server "XXX", has an invalid character "ASCII code XXX". No recorded settings applied.
Invalid transfer server address	The server address is not valid.	Send a syslog warning message including the transfer server address: No configuration file is fetched because the server host "XXX" is invalid. Set <code>sysAdminDownloadConfigFileStatus</code> to <i>fail</i> .
File size too big	Downloading a file with a size exceeding 512000 bytes.	Send a syslog warning message including the file location/name, the transfer server address, the file size and the maximum allowed size: The fetched configuration file "XXX", from server "XXX", has a size "XXX bytes" that exceeds the maximum allowed size "XXX bytes". Set <code>sysAdminDownloadConfigFileStatus</code> to <i>fail</i> .

Table 36: Configuration File Fetching Error Handling (Continued)

Error Type	Cause	Behaviour
Invalid encryption	The configuration file cannot be decrypted. A badly encrypted file is detected if the header or the padding is invalid.	Send a syslog warning message including the file location/name and the transfer server address: The fetched configuration file \"%s\", from server \"%s\", can not be decrypted.
TFTP-Specific Error Handling		
File not found	Received error code 1 (file not found) from the TFTP server.	Send a syslog warning message including the file name and location with the TFTP server address: The configuration file "XXX" was not found on the TFTP server "XXX". Set <i>sysAdminDownloadConfigFileStatus</i> to <i>fail</i> .
Access violation	Received error code 2 (access violation) from the TFTP server.	Send a syslog warning message including the file name and location with the TFTP server address: The configuration file "XXX" was not fetched. There was a TFTP access violation with server "XXX". Set <i>sysAdminDownloadConfigFileStatus</i> to <i>fail</i> .
Connection timeout	No answer from the TFTP server. The time elapsed since the TFTP request was sent exceeds 32 seconds.	Send a syslog warning message including the file name and location with the TFTP server address: The configuration file "XXX" was not fetched. The TFTP connection with server "XXX" timed out. Set <i>sysAdminDownloadConfigFileStatus</i> to <i>fail</i> .
Transfer error	Received a TFTP error (other than error code 1 and 2) from the TFTP server.	Send a syslog warning message including the file name and location with the TFTP server address: Error in the TFTP transfer of the configuration file "XXX" from host "XXX" and port number XXX. Set <i>sysAdminDownloadConfigFileStatus</i> to <i>fail</i> .
File size too big	Downloading a file with a size exceeding 512000 bytes.	Abort the transfer by sending error code 3 (disk full or allocation exceeded) to the TFTP client.
HTTP-Specific Error Handling		
Access unauthorized	Received a 401 Unauthorized from the HTTP server.	Send a syslog warning message including the file location/name with the HTTP server address: The access to configuration file "XXX" is unauthorized on HTTP server "XXX". Set <i>sysAdminDownloadConfigFileStatus</i> to <i>fail</i> .
File not found	Received a 404 Not Found from the HTTP server.	Send a syslog warning message including the file location/name with the HTTP server address: The configuration file "XXX" was not found on the HTTP server "XXX". Set <i>sysAdminDownloadConfigFileStatus</i> to <i>fail</i> .

Table 36: Configuration File Fetching Error Handling (Continued)

Error Type	Cause	Behaviour
Session timeout	No answer from the HTTP server. The time elapsed since the HTTP request was sent exceeds 15 seconds.	Send a syslog warning message including the file location/name with the HTTP server address: The configuration file "XXX" was not fetched. The HTTP session with server "XXX" timed out. Set <code>sysAdminDownloadConfigFileStatus</code> to <code>fail</code> .
Session closed by peer	The HTTP server closed the session.	Send a syslog warning message including the file location/name with the HTTP server address: The configuration file "XXX" HTTP transfer session was closed by peer: host "XXX". Set <code>sysAdminDownloadConfigFileStatus</code> to <code>fail</code> .
Transfer error	Received an HTTP error (other than 401 and 404) from the HTTP server.	Send a syslog warning message including the file location/name with the HTTP server address and port: Error in the HTTP transfer of the configuration file "XXX" from host "XXX" and port number XXX. Set <code>sysAdminDownloadConfigFileStatus</code> to <code>fail</code> .

Firmware Download

This chapter describes how to download from the web interface a firmware version available on the designated firmware server into the Mediatrix 4100.

You have the choice to perform the firmware download by using the TFTP or protocol. You can also configure the Mediatrix 4100 to automatically update its firmware version.

The *Firmware Update* sub-page of the *Management* page allows you to set various firmware download parameters grouped in two categories:

- ▶ General parameters
- ▶ Automatic Update

You can also set these parameters via SNMP, as described in [“Chapter 13 - Software Download” on page 207](#).

Before Downloading

To download a firmware, you may need to setup the following applications on your computer:

- ▶ TFTP server with proper root path
- ▶ MIB browser (with the current Mediatrix 4100 MIB tree)
You can use the MIB browser built in the Media5's Unit Manager Network. See [“Unit Manager Network – Element Management System” on page xxii](#) for more details.
- ▶ Firmware upgrade zip file
- ▶ SNTP server properly configured
- ▶ HTTP server with proper root path
- ▶ Syslog daemon (optional)

Configuring the TFTP Server

If you are to perform a firmware download by using the TFTP protocol, you must install a TFTP (Trivial File Transfer Protocol) server running on the PC designated as the firmware file server. This PC must not have a firewall running. Media5 also recommends to place the PC and the Mediatrix 4100 in the same subnet.

It is assumed that you know how to set the TFTP root path. If not, refer to your TFTP server's documentation.

Configuring the SNTP Server

If you are to use the automatic firmware update feature (see [“Automatic Configuration Update” on page 46](#) for more details), you must have a time server SNTP that is accessible and properly configured. It is assumed that you know how to configure your SNTP server. If not, refer to your SNTP server's documentation. You can also refer to [“Chapter 20 - SNTP Settings” on page 289](#) for more details on how to configure the Mediatrix 4100 for a SNTP server.

Configuring the HTTP Server

If you are to perform a firmware download by using the HTTP protocol, you must install a HTTP server running on the PC designated as the server host. This PC must not have a firewall running. Media5 also recommends to place the PC and the Mediatrix 4100 in the same subnet.

It is assumed that you know how to set the root path. If not, refer to your HTTP server's documentation.

Firmware Servers Configuration

The Mediatrix 4100 must know the IP address and port number of its Primary and Secondary firmware servers. These servers contain the files required for the firmware update. You can assign these information to the Mediatrix 4100 through a DHCP server or manually enter them yourself.

► To set the firmware download server parameters:

1. In the web interface, click the *Management* link, then the *Firmware Download* sub-link.

Figure 26: Management – Firmware Download Web Page

The screenshot shows the Mediatrix 4100 web interface in Microsoft Internet Explorer. The address bar shows <http://10.2.128.37/firmware.esp>. The page has a navigation bar with tabs: Device Info, Management (selected), SIP, Telephony, and Advanced. Under Management, there are sub-tabs: Admin, Network Settings, Configuration File, Firmware Download (selected), and Reboot.

The main content area is titled "Firmware Download" and contains two sections:

- Status:** A table showing the current configuration:

Status	
Firmware Download Server Source:	Static
Firmware Download Primary Server Host:	10.2.121.249
Firmware Download Primary Server Port:	69
Firmware Download Secondary Server Host:	10.2.121.249
Firmware Download Secondary Server Port:	69
Firmware Download Path:	Enterprise\SIP_v5.0.11.72_MX-S5001-02-E
- General:** A form for manual configuration. Red arrows and numbers 2 through 5 point to the following fields:
 - 2: Firmware Download Server Source (radio buttons for Static and DHCP)
 - 3: Firmware Download Protocol (radio buttons for TFTP and HTTP)
 - 4: Firmware Download User Name (text field)
 - 5: Firmware Download Primary Server Host (text field)

Other fields in the General section include Firmware Download Password, Firmware Download Primary Server Port, Firmware Download Secondary Server Host, Firmware Download Secondary Server Port, Firmware Location Provision Source (radio buttons for Static and Remote File), Firmware Location (text field), and Firmware Selection File Location (text field).

2. Select the configuration source of the firmware file information in the *Firmware Download Server Source* choices.

Table 37: Configuration File Information Sources

Source	Description
Static	You manually enter the value and it remains the same every time the Mediatrix 4100 restarts. Use the static configuration if you are not using a DHCP server or if you want to bypass it.
DHCP	The value is obtained at start-time by querying a DHCP server and using standard DHCP fields or options. Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See “Chapter 8 - IP Address and Network Configuration” on page 143 for more details.

3. Set the transfer protocol to use in the *Firmware Download Protocol* field.
You have the choice between **tftp** and **http**.
Your HTTP server may activate some caching mechanism for the firmware download. This mechanism caches the initial firmware download for later processing, thus preventing changes or update of the original download by the user. This can cause problems if a user wants to modify the firmware download and perform it again immediately. The result will still return the original download and not the new one.
4. If your HTTP server requires authentication when downloading the firmware, set the following fields:
 - The user name in the *Firmware Download User Name* field.
 - The password in the *Firmware Download Password* field.
 The Mediatrix 4100 supports basic and digest HTTP authentication, as described in RFC 2617.
5. If the firmware download server configuration source is **Static**:
 - enter the firmware download primary server static IP address or domain name in the *Firmware Download Primary Server Host* field.
 - enter the firmware download primary server static IP port number in the *Firmware Download Primary Server Port* field.
 - enter the firmware download secondary server static IP address or domain name in the *Firmware Download Secondary Server Host* field.
 - enter the firmware download secondary server static IP port number in the *Firmware Download Secondary Server Port* field.
 The default port value complies to RFC 1340 on the well-known ports (assigned numbers). This value applies to a TFTP server. It may be different for other servers. If you are using an HTTP server to perform the firmware download, you must change the port value to 80.
6. Proceed to [“Setting up the Firmware Download” on page 53](#).

Setting up the Firmware Download

Configuration settings are not lost when upgrading the firmware to a newer version. However, configuration settings may be lost if you upload an older firmware to the device. See [“Firmware Downgrade” on page 61](#) for more details.

Extracting the Zip File

The zip file contains the firmware information required for the download.

Extract the contents of the zip file on the PC designated as the firmware download server. Be sure to use the defined folder name. This creates a directory that contains the files required for the Mediatrix 4100 to properly update its firmware.

The directory name must be the same as the name defined in the *Firmware Location* field or *Firmware Selection File Location* field. See [“Setting up the Configuration File Download” on page 42](#) for more details.

Media5 suggests that a folder, named identically to the firmware build, be available and used for the files related to that build only. Each folder should include only one delivery to ensure accuracy.

This directory must be located under the root path as defined in the TFTP/HTTP server or the firmware download will not proceed.

Setting up the Firmware Download Path

When performing a firmware download, you must configure the path, on the remote firmware download server, of the directory where you extracted the files required for the download. This applies to both the manual or automatic download procedure, using the HTTP or TFTP protocol.

The directory must be located under the root path, as defined in the TFTP or HTTP server, or the firmware download will not proceed. See [“Before Downloading” on page 51](#) for more details.

The Mediatrix 4100 first downloads a file called “setup.inf”. This file contains the list of all the other files to download, depending on the product. The “setup.inf” file and all the other files must be in the same directory. If any of the files is missing, the procedure will not work properly.

- To setup the firmware download path:
1. In the *General* section, select where to get the image location in the *Firmware Location Provision Source* field.
You have the following choices:

Table 38: Image Location Parameters

Parameter	Description
static	Uses the directory specified in the <i>Firmware Location</i> field (see Step 2).
remoteFile	The image location is defined in a file called “mediatrix4108targetimage.inf”(the 4108 part may be changed to 4116 or 4124 depending on your product). The location of this file is defined in the <i>imageSelectionFileLocation</i> variable. This is useful if you are using automatic updates with multiple units (see Step 3).

2. If you set the *Firmware Location Provision Source* parameter to **Static** (see Step 1), configure the firmware download path in the *Firmware Location* field.
This is the location of the “setup.inf” file that contains the list of the files to download into the Mediatrix 4100. The “setup.inf” file and all the other files must be in the same directory. In other words, this is the path from the root TFTP/HTTP folder down to the files extracted from the zip file.
Note that the path must contain a maximum of 63 characters.

Figure 27: Management – Firmware Download Web Page

General	
Firmware Download Server Source:	☐ Static ☒ DHCP
Firmware Download Protocol:	☒ TFTP ☐ HTTP
Firmware Download User Name:	
Firmware Download Password:	
Firmware Download Primary Server Host:	chiers.chiers.test.media
Firmware Download Primary Server Port:	69
Firmware Download Secondary Server Host:	192.168.0.10
Firmware Download Secondary Server Port:	69
Firmware Location Provision Source:	☒ Static ☐ Remote File
Firmware Location:	SIP_v5.0.9.64_MX-S500
Firmware Selection File Location:	

3. If you set the *Firmware Location Provision Source* parameter to **remoteFile** (see Step 1):
 - a. Create a text file and write the path and/or name of the directory that contains the files required for download. Save this file as "mediatrix4108targetimage.inf" under the server root path.



Note: If you leave the file empty, the Mediatrix 4100 will look for the firmware download information in the root directory of the image server.

- b. Configure the path of the "mediatrix4108targetimage.inf" file in the *Firmware Selection File Location* field.

Note that the selection file name is in lower case. Some web servers are case sensitive. The path must contain a maximum of 63 characters.

This is useful if you are using automatic updates with multiple units. If you want the units to download a new version, you only have to change the path once in the "mediatrix4108targetimage.inf" file. If you were to use the *Firmware Location* field, you would have to change the path in every unit.

4. Click *Submit* if you do not need to set other parameters.



Note: The current firmware server information is displayed in the *Status* section.

Example

Let's consider the following example:

- ▶ The directory that contains the files required for download is called: **SIP_v5.0.1.1_MX-S5001-01**.
- ▶ This directory is under **C:/Root/Download**.

Table 39: Path Configurations Example

Root Path	Corresponding Path Name
c:/root/download	SIP_v5.0.1.1_MX-S5001-01
c:/	root/download/SIP_v5.0.1.1_MX-S5001-01
c:/root	download/SIP_v5.0.1.1_MX-S5001-01

The following are some tips to help your download process:

- ▶ If available, use the *Browse* button (or equivalent) of the TFTP/HTTP server to select the directory, eliminating typographical errors.
- ▶ Use the "/" character when defining the path to indicate sub-directories. For instance, *root/download*.
If you are using the TFTP protocol to download the firmware, note that some TFTP servers on Windows do not recognize the "/" character and produce an error. In this case, use the "\" character.
- ▶ Use basic directory names, without spaces or special characters such as "~", "@", etc., which may cause problems.
- ▶ Cut and paste the path and/or name of the directory that contains the extracted files into the image path of the Mediatrix 4100 (you may have to convert "\" into "/") to eliminate typographical errors.

Note that you can define the **C:/Root/Download** part as you want. The file names may also differ from the example shown above.

Firmware Download Status

You can validate the status of the firmware download in various ways.

Syslog Messages

If you are using a Syslog daemon, you will receive messages that inform you of the firmware update status. The following are the syslog messages the Mediatrix 4100 sends:

Table 40: Firmware Update Syslog Messages

Level	Message	Event
General Messages		
Informational	The software update succeeded.	The firmware update has been successful.
Error	The software update failed.	The firmware update experienced an error and has not been completed.
Error	The software update failed (xxx).	An error occurs when updating the software, internal error code provided.
Warning	Primary image server not specified, cannot download file: xxx	This error occurs when an image download is initiated and no domain name or address is specified for the primary image server.
Warning	Secondary image server not specified, cannot download file: xxx.	When a request involving the primary server fails, the secondary server is tried. This error occurs when there is no address or domain name specified for the secondary image server.
Error	Cannot resolve address of image server: xxx.	A DNS request failed to resolve the domain name of the image server (primary or secondary).
Error	Target image at location: xxx from host: xxx is invalid or corrupted.	For periodic and automatic updates, the target image to download is first compared with the installed image. This error occurs when this comparison failed because of corruption in the target image files.
Informational	Image download transfer initiated.	When manual, periodic or “at restart” image download is initiated.
Warning	The file: xxx from host: xxx exceeds the size limit.	The selection file or “setup.inf” file received exceeds 10000 bytes.
Informational	Target image at location: xxx from host xxx is identical to currently installed image. Transfer aborted.	For periodic and automatic updates, the target image to download is first compared with the installed image. This message occurs when this comparison determined that the target image is identical to the installed image.
Error	Image does not support hardware (error %d)	The software download failed because the software image is not compatible with the hardware.
HTTP-Specific Messages		
Warning	HTTP image transfer of file: xxx from host: xxx was closed by peer.	The HTTP transfer was closed by the peer.

Table 40: Firmware Update Syslog Messages (Continued)

Level	Message	Event
Warning	HTTP image transfer of file: xxx from host: xxx was closed due to unsupported or malformed response from the host.	In the HTTP response, one of the following error occurred: - The protocol version is not 1.0 or 1.1. - Some field or line is not properly formatted. - The trailing <crLf> is not present at the end of the header. - Unsupported kind of response.
Warning	HTTP image transfer of file: xxx from host: xxx was refused because of a malformed or incompatible request.	When receiving HTTP response #400 or #403.
Warning	HTTP image transfer of file: xxx from host: xxx was refused because of a server error.	When receiving HTTP response #500 or #501.
Warning	HTTP image transfer of file: xxx from host: xxx was refused because service is unavailable.	When receiving HTTP response #503.
TFTP-Specific Messages		
Warning	Image transfer of file: xxx from host: xxx and port: xxx was closed due to unexpected error	Unexpected error, either internal or on a TFTP or HTTP connection.
Warning	Image transfer of file: xxx from host: xxx port: xxx was closed after timeout	When not receiving TFTP packets for 32 seconds or not receiving a HTTP packet for 15 seconds.
Warning	Image transfer. File: xxx not found on host: xxx	When receiving TFTP error "NOT FOUND" or HTTP response #404.
Warning	Image transfer. Access to file: xxx on host: xxx is unauthorized	When receiving TFTP error "ACCESS" or HTTP response #401.

LED States

When the Mediatrix 4100 initiates a firmware download, the LEDs located on the front panel indicate the status of the process.

Table 41: LED States in Firmware Download

Event	LED State
Image downloading and writing	Each LED blinks alternately at 1 Hz with 1/4 ON duty cycle. Warning: Do not turn the Mediatrix 4100 off while in this state.
Image download failed	All LEDs blink at the same time at 2 Hz with 50% ON duty cycle for 4 seconds.

See "[LED Indicators](#)" on page 15 for a detailed description of the LED patterns related to the firmware download process.

Firmware Download Procedure

The following steps explain how to download a firmware from the web interface.

► **To download a firmware version:**

1. If not already done, setup the Image server used to download the firmware (see [“Before Downloading” on page 51](#)).
2. Place the firmware to download on the computer hosting the TFTP or HTTP server.
The file must be in a directory under the TFTP root path.
3. If you are downloading via TFTP, be sure that UDP ports 60000 to 60512 inclusively are opened in your firewall.
4. If not already done, configure the Image path as described in [“Setting up the Configuration File Download” on page 42](#).
5. If not already done, configure the image hosts and ports, as well as the transfer protocol, as defined in [“Firmware Servers Configuration” on page 52](#).



Caution: When downloading via HTTP, the firmware download server's port must be 80. You can see the actual port assigned in the *Status* section of the *Firmware Download* page.

If you are using a DHCP server and it did not provide the proper port, reconfigure it with the proper port or use a static configuration. See [“Configuration File Server Settings” on page 41](#) for more details.

6. Initiate the firmware download by setting the *System Command* drop-down menu of the *Management – Admin* Web Page to **downloadSoftware**.
7. Click *Submit*.
This starts the download process. See [“System Management” on page 33](#) for more details on the system commands.



Caution: Never shutdown the Mediatrix 4100 manually while in the download process, because the image may be partially written and the Mediatrix 4100 is unable to restart.

The firmware download may take several minutes, depending on your Internet connection, network conditions and servers conditions.

If Transparent Address Sharing is enabled during the firmware download, the PC connected to the Mediatrix 4100 may experience momentary loss of Internet connectivity.

Automatic Firmware Update

You can configure the Mediatrix 4100 to automatically update its firmware. This update can be done:

- Every time the Mediatrix 4100 restarts.
- At a specific time interval you can define.

Automatic Update on Restart

The Mediatrix 4100 may download a new firmware each time it restarts.

► **To set the automatic update every time the Mediatrix 4100 restarts:**

1. If not already done, setup the Image server used to download the firmware (see [“Before Downloading” on page 51](#)).
2. Place the firmware to download on the computer hosting the TFTP or HTTP server.
The file must be in a directory under the TFTP root path.
3. If you are downloading via TFTP, be sure that UDP ports 60000 to 60512 inclusively are opened in your firewall.

4. If not already done, configure the Image path as described in [“Setting up the Configuration File Download” on page 42](#).
5. If not already done, configure the image hosts and ports, as well as the transfer protocol, as defined in [“Firmware Servers Configuration” on page 52](#).



Caution: When downloading via HTTP, the firmware download server's port must be 80. You can see the actual port assigned in the *Status* section of the *Firmware Download* page.

If you are using a DHCP server and it did not provide the proper port, reconfigure it with the proper port or use a static configuration. See [“Configuration File Server Settings” on page 41](#) for more details.

6. In the *Automatic Update* section of the *Firmware Download* page, select **Enable** in the *Firmware Download On Restart* field.

Figure 28: Management – Automatic Update section

The automatic firmware download will be performed each time the Mediatrix 4100 restarts.

7. Click *Submit* if you do not need to set other parameters.

Automatic Update at a Specific Time Interval

You can configure the Mediatrix 4100 to download a new firmware at a specific day and/or time.

► To set the automatic update at a specific time interval:

1. If not already done, setup the Image server used to download the firmware (see [“Before Downloading” on page 51](#)).
2. Place the firmware to download on the computer hosting the TFTP or HTTP server.
The file must be in a directory under the TFTP root path.
3. If you are downloading via TFTP, be sure that UDP ports 60000 to 60512 inclusively are opened in your firewall.
4. If not already done, configure the Image path as described in [“Setting up the Configuration File Download” on page 42](#).
5. If not already done, configure the image hosts and ports, as well as the transfer protocol, as defined in [“Firmware Servers Configuration” on page 52](#).



Caution: When downloading via HTTP, the firmware download server's port must be 80. You can see the actual port assigned in the *Status* section of the *Firmware Download* page.

If you are using a DHCP server and it did not provide the proper port, reconfigure it with the proper port or use a static configuration. See [“Configuration File Server Settings” on page 41](#) for more details.

6. In the *Automatic Update* section of the *Firmware Download* page, select **Enable** in the *Firmware Download Periodic Update* field.

Figure 29: Management – Automatic Update section

7. Set the waiting period between each firmware update in the *Periodic Update Period* field.
The time unit for the period is specified in the *Periodic Update Time Unit* field (see Step 6). Available values are from 1 to 48.
8. Define the time base for automatic firmware updates in the *Periodic Update Time Unit* field.
You have the following choices:

Table 42: Time Unit Parameters

Parameter	Description
Minutes	Updates the unit's firmware every x minutes. You can specify the x value in the <i>Periodic Update Period</i> field (see Step 5).
Hours	Updates the unit's firmware every x hours. You can specify the x value in the <i>Periodic Update Period</i> field (see Step 5).
Days	Updates the unit's firmware every x days. You can specify the x value in the <i>Periodic Update Period</i> field (see Step 5). You can also define the time of day when to perform the update in the <i>Periodic Update Time Range</i> field (see Step 7).

9. If you have selected **Days** in Step 6, set the time of the day when to initiate a firmware update in the *Periodic Update Time Range* field.
The time of the day is based on the *SNTP Timezone* field of the *Management - Network Settings* page (see [“SNTP Settings” on page 37](#) for more details).
You must have a time server SNTP that is accessible and properly configured, or the automatic firmware update feature may not work properly. It is assumed that you know how to configure your SNTP server. If not, refer to your SNTP server's documentation. You can also refer to [“SNTP Settings” on page 37](#) for more details on how to configure the Mediatrix 4100 for a SNTP server.
If a time range is specified, the unit will initiate the image software download at a random time within the interval specified.

The format should be one of the following:

```
hh[:mm[:ss]]
hh[:mm[:ss]] - hh[:mm[:ss]]
```

Where:

hh: Hours.
mm: Minutes.
ss: Seconds.

The image software download is initiated at the first occurrence of this value and thereafter with a period defined by the *Periodic Update Period* field. Let's say for instance the automatic update is set with the time of day at 14h00 and the update period at every 2 days.

- If the automatic update is enabled before 14h00, the first update will take place the same day at 14h00, then the second update two days later at the same hour, and so on.

- If the time range is set to '14:00 - 15:00' and the automatic update is enabled within those hours, the first update will take place the following day. This means that a range of '00:00:00 - 23:59:59' will always take place the next day.



Note: The *Periodic Update Time of Day* field is deprecated. It is recommended to use the *Periodic Update Time Range* field because it has precedence over this field.

10. Click *Submit* if you do not need to set other parameters.

Spanning Tree Protocol (STP)

Many network switches use the Spanning Tree Protocol (STP) to manage Ethernet ports activity. When a firmware download occurs, the *Ethernet* connector of the Mediatrix 4100 may switch off. This shutdown may trigger these network switches to shutdown the matching Ethernet port for at least one minute. This shutdown on the switch side can prevent firmware download.

To prevent this, the Mediatrix 4100 supports the STP. However, this management has a potential time cost. It may appear from time to time that firmware downloads take more time. This is normal.

The following is an example where the STP management impacts the download duration.

- ▶ The firmware download procedure does not use any DHCP and DNS services.
- ▶ The primary image server is down (or not properly configured).
- ▶ The secondary image server is up and running well.

In this case, the Mediatrix 4100 tries to contact the primary image server. As it is not available, the Mediatrix 4100 retries for two minutes. It contacts the secondary server after that period and starts the firmware download.



Note: When using the Mediatrix 4100, Media5 recommends to disable the Spanning Tree Protocol on the network to which the unit is connected.

Firmware Downgrade

It is possible to downgrade a Mediatrix 4100 from the current version (for instance, v5.0rx.x) to an older version (for instance, v4.4rx.x).



Note: If you perform a default reset on the Mediatrix 4100, you must download the current version into the unit before performing the firmware downgrade procedure.

▶ To perform a firmware downgrade:

1. Create, in a common folder under the TFTP root path, the current (for instance, v5.0) and older (for instance, v4.4) applications folders.
2. Re-update the Mediatrix 4100 with the current application.
The Mediatrix 4100 runs the current firmware version (v5.0rx.x).
3. Perform the firmware downgrade to the older application (v4.4rx.x) as described in ["Firmware Download Procedure" on page 58](#).

Emergency Firmware Procedure

If the firmware download is suddenly interrupted, it may not be complete. Without any protection against this situation, the Mediatrix 4100 is not functional.

A transfer may be interrupted for the following reasons:

- ▶ An electrical shortage.
- ▶ The user of the Mediatrix 4100 can accidentally power off the unit.

Depending on the moment when the firmware download has been interrupted, the emergency firmware procedure (also called rescue application) can automatically start a new firmware download to repair the firmware if it has been corrupted by the interruption. However, there is a small but critical time frame during which unrecoverable errors could happen. This is why it is very important that the unit is not turned off during firmware downloads.

Using the Emergency Firmware

When the emergency firmware procedure starts, the following steps apply:

1. The Mediatrix 4100 tries to initiate the firmware download with the primary firmware server.
2. If the firmware download fails with the primary firmware server, the Mediatrix 4100 tries to initiate the firmware download with the secondary firmware server.
3. If the primary and the secondary servers cannot be reached, the Mediatrix 4100 tries two default servers: 192.168.0.10 and then 192.168.0.2.
If, for some reason, it is impossible to rescue the unit by using the primary and secondary servers, setting up a server at one of these addresses within the correct subnet will provide an ultimate way to rescue the unit. However, if these addresses cannot be reached from the unit's subnet, the default gateway must provide appropriate routing to them.
4. If the firmware download also fails with the two default servers, the Mediatrix 4100 idles for one minute.
5. After this one minute, the Mediatrix 4100 tries to initiate the firmware download again.
6. If the firmware download fails again with the primary, secondary, and default firmware servers, the Mediatrix 4100 idles for two minutes before attempting to initiate the firmware download.
7. If the emergency firmware download still fails, the Mediatrix 4100 tries to initiate the firmware download again by doubling the delay between each attempt up to a maximum of 16 minutes:
 - first attempt: 1 minute delay
 - second attempt: 2 minutes delay
 - third attempt: 4 minutes delay
 - fourth attempt: 8 minutes delay
 - fifth attempt: 16 minutes delay
 - sixth attempt: 16 minutes delay
 - etc.

This procedure continues until the firmware download completes successfully. The firmware download can fail if the firmware server cannot be reached or if the firmware directory is not found on the firmware server.

Web Interface – SIP Parameters

The *SIP* page allows you to configure the various SIP-related parameters of the Mediatrix 4100:

- ▶ General SIP configuration parameters
- ▶ SIP Interop parameters
- ▶ Authentication parameters

SIP Servers Configuration

Standards Supported

- RFC 3903 – Session Initiation Protocol (SIP) Extension for Event State Publication
- RFC 3863 – Presence Information Data Format (PIDF)

The *Configuration* sub-page of the *SIP* page allows you to configure the SIP server and SIP user agent parameters of the Mediatrix 4100.

SIP Servers

The Mediatrix 4100 uses the following types of servers:

- ▶ Registrar server
- ▶ Proxy server
- ▶ Outbound Proxy server
- ▶ Presence Compositor server

Registrar Server

The registrar server accepts REGISTER requests and places the information it receives in those requests into the location service for the domain it handles.

Proxy Server

The proxy server is an intermediary program that acts as both a server and a client for the purpose of making requests on behalf of other clients. A proxy server primarily plays the role of routing, which means its job is to ensure that a request is passed on to another entity that can further process the request. Proxies are also useful for enforcing policy and for firewall traversal. A proxy interprets, and, if necessary, rewrites parts of a request message before forwarding it.

Outbound Proxy Server

An outbound proxy is an intermediary entity that acts as both a server and a client for the purpose of making requests on behalf of other clients. The outbound proxy receives all outbound traffic and forwards it. Incoming traffic may or may not go through the outbound proxy. The outbound proxy's address is never used in the SIP packets, it is only used as a physical network destination for the packets.

When the outbound proxy is enabled, the proxy is still used to create the *To* and *From* headers, but the packets are physically sent to the outbound proxy.

The outbound proxy is enabled if the IP address is valid (i.e., not 0.0.0.0).

Presence Compositor Server

A User Agent Server (UAS) that processes PUBLISH requests and is responsible for composing event state into a complete, composite event state of a resource for a presentity.

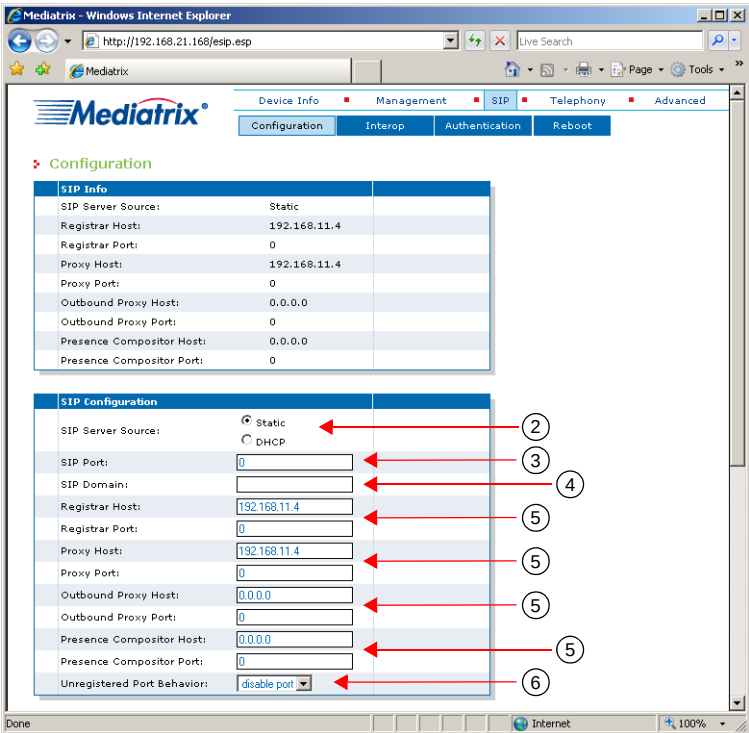
SIP Configuration

The Mediatrix 4100 must know the IP address and port number of the SIP servers. You can assign these information to the Mediatrix 4100 through a DHCP server or manually enter them yourself.

You can also set these parameters via SNMP, as described in [“Chapter 9 - SIP Servers” on page 165](#) and [“Chapter 18 - SIP Protocol Features” on page 259](#).

- To set the SIP servers configuration:
1. In the web interface, click the *SIP* link, then the *Configuration* sub-link.

Figure 30: SIP – Configuration Web Page



2. Select the configuration source of the SIP servers information in the *SIP Server Source* choices.

Table 43: SIP Servers Configuration Sources

Source	Description
Static	You manually enter the value and it remains the same every time the Mediatrix 4100 restarts. Use the static configuration if you are not using a DHCP server or if you want to bypass it.
DHCP	The value is obtained at start-time by querying a DHCP server and using standard DHCP fields or options. Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See “Chapter 8 - IP Address and Network Configuration” on page 143 for more details.

3. Set the user agent port number in the *SIP Port* field.
If this field is set to 0, the default SIP port is used.

4. Define whether or not to override the default proxy home domain used by entering a domain in the *SIP Domain* field.

This value replaces the home domain proxy host as defined in the *Proxy Host* field. It is used by the address of record in the *To* and *From* headers.

5. If the SIP server configuration source is **Static**:
 - enter the SIP registrar server static IP address or domain name in the *Registrar Host* field.
 - enter the SIP registrar server static IP port number in the *Registrar Port* field.
 - enter the SIP Proxy server static IP address or domain name in the *Proxy Host* field.
 - enter the SIP Proxy server static IP port number in the *Proxy Port* field.
 - enter the SIP outbound proxy server static IP address or domain name in the *Outbound Proxy Host* field.
Setting the address to **0.0.0.0** disables the outbound proxy.
 - enter the SIP outbound proxy server static IP port number in the *Outbound Proxy Port* field.



Note: If the port number corresponds to a domain name that is bound to a SRV record, the port must be set to **0** for the unit to perform DNS requests of type SRV (as per RFC 3263). Otherwise, the unit will not use DNS SRV requests, but will rather use only requests of type A because it does not need to be specified which port to use.

- enter the SIP Presence Compositor server static IP address or domain name in the *Presence Compositor Host* field.
 - enter the SIP Presence Compositor server static IP port number in the *Presence Compositor Port* field.
6. Specify whether a line should remain enabled or not when not registered in the *Unregistered Port Behavior* field.
This is useful if you want your users to be able to make calls even if the line is not registered with the SIP server. The following values are available:

Table 44: Unregistered Line Behaviour

Value	Description
disable port	When the line is not registered, it is disabled. The user cannot make or receive calls. Picking up the handset yields a fast busy tone, and incoming INVITEs receive a "403 Forbidden" response.
enable port	When the line is not registered, it is still enabled. The user can receive and initiate outgoing calls. Note that because the line is not registered to a registrar, its public address is not available to the outside world; the line will most likely be unreachable except through direct IP calling.

7. Click *Submit* if you do not need to set other parameters.
The current SIP server information is displayed in the *SIP Info* section.

SIP User Agent

A user agent is a logical entity that can act as both client and server for the duration of a dialog. Each line (also known as endpoint) of the Mediatrix 4100 is a user agent.

You can set information for each user agent such as its telephone number and friendly name. This information is used to dynamically create the *To*, *From* and *Contact* headers used in the request the user agent sends. These headers make up the caller ID information that is displayed on telephones/faxes equipped with a proper LCD display.

► To set user agent information:

1. In the second *SIP Configuration* section of the *SIP* page, enter a user name for each port in the *User Name* column.

Figure 31: SIP – User Agent section

Port	User Name	Friendly Name	Other Accepted Username
1	3330001		
2	3330002		
3	3330003		
4	3330004		

The user name uniquely identifies this endpoint in the domain, such as a telephone number. It is used to create the *Contact* and *From* headers. The *From* header carries the permanent location (IP address, home domain) where the endpoint is located. The *Contact* header carries the current location (IP address) where the endpoint can be reached. Contact headers are used in two ways:

- First, contacts are registered to the registrar. This enables callers to be redirected to the endpoint's current location.
- Second, a contact header is sent along with any request the user agent sends (e.g., INVITE), and is used by the target user agent as a return address for later requests to this endpoint.

You cannot set this field to an empty value. Furthermore, it is reset to 333000X during a factory reset, the X digit being the port number.

2. Enter another name for each line in the *Display Name* column.

This is a friendly name for the user agent. It contains a descriptive version of the URI and is intended to be displayed to a user interface.

3. Enter a second accepted user name for each line in the *Other Accepted Username* field.

This is a user name that the endpoint recognizes as its own, but does not register in contacts sent to the registrar. The endpoint only registers the user name set in the *User Name* column.

You can use this column to add a variation on the user name. For instance, let's say that the user name is a telephone number, 555-1111. A variation could be to prefix the local area or country code, such as 819-555-1111.

To include more than one user name, separate them with a "," character, such as: user1, user2, 5552222, 18195552222.

4. Click *Submit* if you do not need to set other parameters.

SIP Registration

You can refresh the registration, i.e., commit the changes you have done to the registration. When refreshing the registration, all enabled endpoints unregister themselves from the previous registrar and send a new registration to the current registrar with the current parameters.

You can also set this parameter via SNMP, as described in ["Registration Parameters" on page 268](#).

► To refresh the registrations:

1. In the *SIP Registration* section of the *SIP* page, set the registration command in the *SIP Registration Command* menu.

Figure 32: SIP – SIP Registration section

The following values are available:

- noOp: No operation.
- refresh: Refresh registrations.

SIP Publication

You can refresh the publication, i.e., commit the changes you have done to the publication. When refreshing the publications, all enabled endpoints unpublish themselves from the previous Presence Compositor and send a new publication to the current Presence Compositor with the current parameters.

You can also set this parameter via SNMP, as described in [“Publication Parameters” on page 269](#).

► To refresh the publications:

1. In the *SIP Publication* section of the *SIP* page, set the publication command in the *SIP Publication Command* menu.

Figure 33: SIP – SIP Publication section



The following values are available:

- noOp: No operation.
- refresh: Refresh publications.

SIP Interop

The *Interop* sub-page of the *SIP* page allows you to configure the SIP penalty box, SIP transport parameters, and specific interop parameters of the Mediatrix 4100.

SIP Penalty Box

The penalty box feature is used to “quarantine” a given host which address times out. During that time, the address is considered as “non-responding” for all requests.

This feature is most useful when using multiple servers and some of them are down. It ensures that users wait a minimal period of time before trying a secondary host.

You can also set these parameters via SNMP, as described in [“SIP Penalty Box” on page 267](#).

Penalty Box vs Transport Types

Media5 recommends to use this feature with care when supporting multiple transports (see [“SIP Transport Type” on page 68](#) for more details) or you may experience unwanted behaviours.

When the Mediatrix 4100 must send a packet, it retrieves the destination from the packet. If the destination address does not specify a transport to use and does not have a DNS SRV entry that configures which transport to use, then the Mediatrix 4100 tries all transports it supports, starting with UDP. If this fails, it tries with TCP. The unit begins with UDP because all SIP implementations must support this transport, while the mandatory support of TCP was only introduced in RFC 3261.



Note: An important fact is that it is not the destination itself that is placed in the penalty box, but the combination of address, port and transport. When a host is in the penalty box, it is never used to try to connect to a remote host unless it is the last choice for the Mediatrix 4100 and there are no more options to try after this host.

Let's say for instance that the Mediatrix 4100 supports both the UDP and TCP transports. It tries to reach endpoint “B” for which the destination address does not specify a transport and there is no DNS SRV entry to specify which transports to use in which order. It turns out that this endpoint “B” is also down. In this case, the Mediatrix 4100 first tries to contact endpoint “B” via UDP. After a timeout period, UDP is placed in the penalty box and the unit then tries to contact endpoint “B” via TCP. This fails as well and TCP is also placed in the penalty box.

Now, let's assume endpoint "B" comes back to life and the Mediatix 4100 tries again to contact it before UDP and TCP are released from the penalty box. First, the unit tries UDP, but it is currently in the penalty box and there is another transport left to try. The Mediatix 4100 skips over UDP and tries the next target, which is TCP. Again, TCP is still in the penalty box, but this time, it is the last target the Mediatix 4100 can try, so penalty box or not, TCP is used all the same to try to contact endpoint "B".

There is a problem if endpoint "B" only supports UDP (RFC 2543-based implementation). Endpoint "B" is up, but the Mediatix 4100 still cannot contact it: with UDP and TCP in the penalty box, the unit only tries to contact endpoint "B" via its last choice, which is TCP.

The same scenario would not have any problem if the penalty box feature was disabled. Another option is to disable TCP in the Mediatix 4100, which makes UDP the only possible choice for the unit and forces to use UDP even if it is in the penalty box.

You must fully understand the above problem before configuring this feature. Mixing endpoints that do not support the same set of transports with this feature enabled can lead to the above problems, so it is suggested to either properly configure SRV records for the hosts that can be reached or be sure that all hosts on the network support the same transport set before enabling this feature.

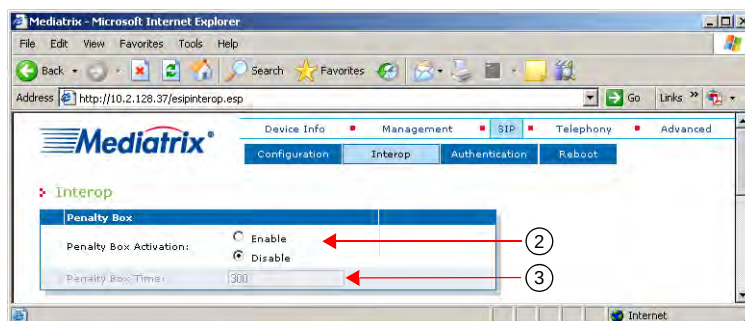
Penalty Box Configuration

The following steps describe how to configure the penalty box feature.

► To set the SIP penalty box parameters:

1. In the web interface, click the *SIP* link, then the *Interop* sub-link.

Figure 34: SIP – Interop Web Page



2. In the *Penalty Box* section, enable the SIP penalty box feature by selecting **Enable** in the *Penalty Box Activation* choices.

The penalty box is always "active". This means that even if the feature is disabled, IP addresses are marked as invalid, but they are still tried. This has the advantage that when the feature is enabled, IP addresses that were already marked as invalid are instantly put into the penalty box.

3. Set the amount of time, in seconds, that a host spends in the penalty box in the *Penalty Box Time* field.

Changing the value does not affect IP addresses that are already in the penalty box. It only affects new entries in the penalty box.

4. Click *Submit* if you do not need to set other parameters.

SIP Transport Type

Standards Supported

RFC 3261 – SIP: Session Initiation Protocol

You can globally set the transport type for all the lines of the Mediatix 4100 to either UDP (User Datagram Protocol) or TCP (Transmission Control Protocol). The Mediatix 4100 will include its supported transports in its registrations.

Please note that RFC 3261 states the implementations must be able to handle messages up to the maximum datagram packet size. For UDP, this size is 65,535 bytes, including IP and UDP headers. However, the maximum datagram packet size the Mediatix 4100 supports for a SIP request or response is 5120 bytes excluding the IP and UDP headers. This should be enough, as a packet is rarely bigger than 2500 bytes.

You can also set these parameters via SNMP, as described in [“SIP Transport Type” on page 265](#).

► **To set the SIP transport type parameters:**

1. In the *SIP Transport* section of the *Interop* page, enable the transport types to use in the proper *Activation* drop-down menu.

You can enable or disable the **UDP** and **TCP** transports.

Figure 35: SIP Transport Section

SIP Transport	Protocol	Activation	QValue
UDP		Enable	
TCP		Disable	

2. Set the priority order of the UDP and TCP transports in the proper *Q Value* field.
A qvalue parameter is added to each contact. The qvalue gives each transport a weight, indicating the degree of preference for that transport. A higher value means higher preference.
The format of the qvalue string must follow the RFC 3261 ABNF (a floating point value between 0.000 and 1.000). If you specify an empty string, no qvalue is set in the contacts.
3. Click *Submit* if you do not need to set other parameters.

Interop Parameters

The interop parameters allow the Mediatix 4100 to properly work, communicate, or connect with specific IP devices.

► **To set interop parameters:**

1. In the *Interop* section of the *Interop* page, set the *Escape Pound (#) in SIP URI Username* drop-down menu with the proper behaviour.

This allows you to define whether or not the pound character (#) must be escaped in the username part of a SIP URI.

Figure 36: Interop Section

Escape Pound (#) in SIP URI Username:	Enable	1
Allow Media Reactivation In Answer:	Disable	2
Allow Audio And Image Negotiation:	Disable	3
Transfer Version:	Transfer 05 Using Refer 02	4

Table 45: Escaping Pound Character Parameter

Parameter	Description
Enable	The Pound character (#) is escaped in the username part of a SIP URI.
Disable	The Pound character (#) is not escaped in the username part of a SIP URI. Note that RFC 3261 specifies that the pound character (#) needs to be escaped in the username part of a SIP URI.

2. Set the *Allow Media Reactivation in Answer* drop-down menu with the behaviour of the Mediatix 4100 when receiving a SDP answer activating a media that had been previously deactivated in the offer

Table 46: Media Reactivation Parameters

Parameter	Description
Enable	A media reactivated in an incoming answer is ignored. This behaviour goes against the SDP Offer/Answer model described by IETF RFC 3264.
Disable	A media reactivated in an incoming answer ends the current media negotiation and the call. This behaviour follows the SDP Offer/Answer model described by IETF RFC 3264.

3. Set the *Allow Audio and Image Negotiation* drop-down menu with the behaviour of the Mediatix 4100 when offering media or answering to a media offer with audio and image negotiation

Table 47: Audio and Image Negotiation Parameters

Parameter	Description
Enable	The unit offers audio and image media simultaneously in outgoing SDP offers and transits to T.38 mode upon reception of a T.38 packet. Also, when the unit answers positively to a SDP offer with audio and image, it transits to T.38 mode upon reception of a T.38 packet.
Disable	Outgoing offers never include image and audio simultaneously. Incoming offers with audio and image media with a non-zero port are considered as offering only audio.

4. Set the *Transfer Version* drop-down with the proper transfer version.

Table 48: Call Transfer Versions Supported

Version	Description
Transfer 02	The Mediatix 4100 executes transfers by using the methods described in the now expired <i>draft-ietf-sip-cc-transfer-02.txt</i> . Its use is deprecated and you should use this setting for backward compatibility issues only.
Transfer 05 Using Refer 02	The Mediatix 4100 executes transfers by using the methods described in the more recent <i>draft-ietf-sip-cc-transfer-05.txt</i> . This draft version contains several enhancements over the previous ones. Among others, it is possible to use the <i>Replaces</i> header to provide a more seamless attended transfer to the user. This method also uses <i>draft-ietf-sip-refer-02.txt</i> . Use this setting if you do not need to interop with transfer02-enabled parties. See "Replaces Configuration Setting" on page 272 for more details.
Sipping Transfer 01 Using Refer RFC 3515	The Mediatix 4100 executes transfers by using the methods described in <i>draft-ietf-sipping-cc-transfer-01.txt</i> . This draft version is more recent than Transfer 02 and Transfer 05 Using Refer 02. This method also uses the <i>RFC 3515 - The Session Initiation Protocol (SIP) Refer Method</i> .

5. Click *Submit* if you do not need to set other parameters.

SIP Authentication

The *Authentication* sub-page of the *SIP* page allows you to configure the unit and user agent authentication parameters of the Mediatrix 4100.

Standards Supported	Basic and Digest authentication as per RFC 3261
----------------------------	---

Authentication information allows you to add some level of security to the Mediatrix 4100 lines by setting user names and passwords. You can add two types of authentication information:

- ▶ **user agent specific authentication**
You can define up to five user names and five passwords for each user agent of the Mediatrix 4100. A user agent can thus register with five different realms.
- ▶ **unit authentication**
You can define up to five user names and five passwords for the Mediatrix 4100. These user names and passwords apply to all lines of the unit.

When a realm requests authentication, the user agent specific authentication is tried first, and then the unit authentication if required.

You can also set these parameters via SNMP, as described in [“Authentication” on page 262](#).

▶ **To set the SIP security parameters:**

1. In the web interface, click the *SIP* link, then the *Authentication* sub-link.

Figure 37: SIP – Authentication Web Page

2. In the *Unit Authentication* section, select whether or not the current unit credentials are valid for any realm in the corresponding *Validate Realm* drop-down menu.

Table 49: Realm Authentication Parameters

Parameter	Description
Disable	The current unit credentials are valid for any realm. The corresponding <i>Realm</i> field is read-only and cannot be modified.

Table 49: Realm Authentication Parameters (Continued)

Parameter	Description
Enable	The unit credentials are used only for a specific realm set in the corresponding <i>Realm</i> field.

3. Enter a realm for each authentication row in the *Realm* column.
When authentication information is required from users, the realm identifies who requested it.
4. Enter a string that uniquely identifies this endpoint in the realm in the *Username* column.
5. Enter a user password in the *Password* column.
6. In the *User Agent Authentication* section, select whether or not the current user agent credentials are valid for any realm in the corresponding *Validate Realm* drop-down menu.

Table 50: Realm Authentication Parameters

Parameter	Description
Disable	The current user agent credentials are valid for any realm. The corresponding <i>Realm</i> field is read-only and cannot be modified.
Enable	The user agent credentials are used only for a specific realm set in the corresponding <i>Realm</i> field.

7. Enter up to five realms for each user agent in the *Realm* column.
When authentication information is required from users, the realm identifies who requested it.
8. Enter a string that uniquely identifies this user agent in the realm in the *Username* column.
9. Enter a user password in the *Password* column.
10. Click *Submit* if you do not need to set other parameters.

The *Telephony* page allows you to configure the various telephony parameters of the Mediatrix 4100.

Digit Maps

Standards Supported

RFC 2705 – Media Gateway Control Protocol (MGCP) Version 1.0, section 3.4 (Formal syntax description of the protocol).

A digit map allows you to compare the number users just dialed to a string of arguments. If they match, users can make the call. If not, users cannot make the call and get an error signal. It is thus essential to define very precisely a digit map before actually implementing it, or your users may encounter calling problems.

Because the Mediatrix 4100 cannot predict how many digits it needs to accumulate before transmission, you could use the digit map, for instance, to determine exactly when there are enough digits entered from the user to place a call.

Syntax

The permitted digit map syntax is taken from the core MGCP specification, RFC 2705, section 3.4:

```
DigitMap = DigitString / '(' DigitStringList ')'
DigitStringList = DigitString 0* ( '|' DigitString )
DigitString = 1*(DigitStringElement)
DigitStringElement = DigitPosition ['.']
DigitPosition = DigitMapLetter / DigitMapRange
DigitMapLetter = DIGIT / '#' / '*' / 'A' / 'B' / 'C' / 'D' / 'T'
DigitMapRange = 'x' / '[' 1*DigitLetter ']'
DigitLetter ::= *((DIGIT '-' DIGIT) / DigitMapLetter)
```

Where “x” means “any digit” and “.” means “any number of”.

For instance, using the telephone on your desk, you can dial the following numbers:

Table 51: Number Examples

Number	Description
0	Local operator
00	Long distance operator
xxxx	Local extension number
8xxxxxxx	Local number
#xxxxxxx	Shortcut to local number at other corporate sites
91xxxxxxxxxx	Long distance numbers
9011 + up to 15 digits	International number

The solution to this problem is to load the Mediatrix 4100 with a digit map that corresponds to the dial plan.

A Mediatrix 4100 that detects digits or timers applies the current dial string to the digit map, attempting a match to each regular expression in the digit map in lexical order.

- ▶ If the result is under-qualified (partially matches at least one entry in the digit map), waits for more digits.
- ▶ If the result matches, dials the number.
- ▶ If the result is over-qualified (i.e., no further digits could possibly produce a match), sends a fast busy signal.

Special Characters

Digit maps use specific characters and digits in a particular syntax. Those characters are:

Table 52: Digit Map Characters

Character	Use
Digits (0, 1, 2... 9)	Indicates specific digits in a telephone number expression.
T	The Timer indicates that if users have not dialed a digit for the time defined, it is likely that they have finished dialing and the SIP Server can make the call.
x	Matches any digit, excluding “#” and “*”.
	Indicates a choice of matching expressions (OR).
.	Matches an arbitrary number of occurrences of the preceding digit, including 0.
[	Indicates the start of a range of characters.
]	Indicates the end of a range of characters.

How to Use a Digit Map

Let's say you are in an office and you want to call a co-worker's 3-digits extension. You could build a digit map that says “after the user has entered 3 digits, make the call”. The digit map could look as follows:

xxx

You could refine this digit map by including a range of digits. For instance, you know that all extensions in your company either begin with 2, 3, or 4. The corresponding digit map could look as follows:

[2-4]xx

If the number you dial begins with anything other than 2, 3, or 4, the call is not placed and you get a busy signal.

Combining Several Expressions

You can combine two or more expressions in the same digit map by using the “|” operator, which is equal to OR.

Let's say you want to specify a choice: the digit map is to check if the number is internal (extension), or external (a local call). Assuming that you must first dial “9” to make an external call, you could define a digit map as follows:

([2-4]xx|9[2-9]xxxxxx)

The digit map checks if:

- ▶ the number begins with 2, 3, or 4 **and**
- ▶ the number has 3 digits

If not, it checks if:

- ▶ the number begins with 9 **and**
- ▶ the second digit is any digit between 2 and 9 **and**
- ▶ the number has 7 digits



Note: Enclose the digit map in parenthesis when using the “|” option.

Using the # and * Characters

It may sometimes be required that users dial the “#” or “*” to make calls. This can be easily incorporated in a digit map:

```
xxxxxxx#
xxxxxxx*
```

The “#” or “*” character could indicate users must dial the “#” or “*” character at the end of their number to indicate it is complete. You can specify to remove the “#” or “*” found at the end of a dialed number. See [“General Parameters” on page 76](#).

Using the Timer

You can configure the Timer. See [“General Parameters” on page 76](#) for more details. It indicates that if users have not dialed a digit for the time defined, it is likely that they have finished dialing and the Mediatrix 4100 can make the call. A digit map for this could be:

```
[2-9]xxxxxxT
```



Note: When making the actual call and dialing the number, the Mediatrix 4100 automatically removes the “T” found at the end of a dialed number, if there is one (after a match). This character is for indication purposes only.

Calls Outside the Country

If your users are making calls outside their country, it may sometimes be hard to determine exactly the number of digits they must enter. You could devise a digit map that takes this problem into account:

```
001x.T
```

In this example, the digit map looks for a number that begins with 001, and then any number of digits after that (x.).

Example

[Table 51 on page 73](#) outlined various call types one could make. All these possibilities could be covered in one digit map:

```
(0T|00T|[1-7]xxx|8xxxxxxx|#xxxxxxx|91xxxxxxxxxxx|9011x.T)
```

Validating a Digit Map

The Mediatrix 4100 validates the digit map as you are entering it and it forbids any invalid value.

General Parameters

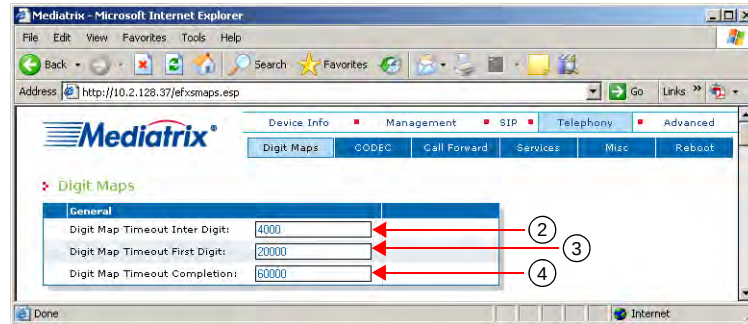
The following are the general digit maps parameters you can set.

You can also set these parameters via SNMP, as described in [“Chapter 21 - Digit Maps” on page 293](#).

► **To set the general digit map parameters:**

1. In the web interface, click the *Telephony* link, then the *Digit Maps* sub-link.

Figure 38: Telephony – Digit Maps Web Page



2. In the *General* section, define the value of the “T” digit in the *Digit Map Timeout Inter Digit* field. The “T” digit expresses a time lapse between the detection of two DTMFs. This value is expressed in milliseconds (ms). Values range from 500 ms to 10000 ms.
3. Define the time between the start of the dial tone and the receiver off-hook tone, if no DTMF is detected, in the *Digit Map Timeout First Digit* field. This value is expressed in milliseconds (ms). Values range from 1000 ms to 180000 ms.
4. Define the total time the user has to dial the DTMF sequence in the *Digit Map Timeout Completion* field. The timer starts when the dial tone is played. When the timer expires, the receiver off-hook tone is played. This value is expressed in milliseconds (ms). Values range from 1000 ms to 180000 ms.
5. Click *Submit* if you do not need to set other parameters.

Allowed Digit Maps

You can create/edit ten digit maps for the Mediatrix 4100. Digit map rules are checked sequentially. If a telephone number potentially matches two of the rules, the first rule encountered is applied.

► To set up digit maps:

1. In the *Allowed Digit Map* section – *Activation* column, enable one or more digit maps by selecting the corresponding **Enable** choice.

Figure 39: Allowed Digit Map Section

Index	Activation	Digit Map	Remove Prefix	Add Prefix	Remove Suffix	Line To Apply
1	☒ Enable ☐ Disable	91	3	6142998		all
2	☒ Enable ☐ Disable	k.T	0			all
3	☒ Enable ☐ Disable	k.T	0			all
4	☒ Enable ☐ Disable	k.T	0			all
5	☒ Enable ☐ Disable	k.T	0			all
6	☒ Enable ☐ Disable	k.T	0			all
7	☒ Enable ☐ Disable	k.T	0			all
8	☒ Enable ☐ Disable	k.T	0			all
9	☒ Enable ☐ Disable	k.T	0			all
10	☒ Enable ☐ Disable	k.T	0			all

2. Define the digit map string that is considered valid when dialed in the *Digit Map* column.
The string must use the syntax described in [“Digit Maps” on page 73](#). A digit map string may have a maximum of 64 characters.
3. Define the amount of digits to remove from the beginning of the dialed number, after dialing, but before initiating the call, in the *Remove Prefix* column.
For instance, when dialing “1-819-xxx-xxxx”, specifying a value of “4” means that the call is started by using the number “xxx-xxxx”.
This rule is applied BEFORE applying both the *Add Prefix* (Step 4) and *Remove Suffix* (Step 5) rules.
4. Define the string to insert at the beginning of the dialed number before initiating the call in the *Add Prefix* column.
For instance, let’s say that you need to dial a special digit, “9”, for all local calls. Dialing “xxx-xxxx” with a value of “9” would yield “9-xxx-xxxx” as the number with which to initiate the call.
This rule is applied AFTER applying both the *Remove Prefix* (Step 3) and *Remove Suffix* (Step 5) rules.
5. Define the string to look for and remove, from the end of the dialed number, in the *Remove Suffix* column.
This is helpful if one of the digit maps contains a terminating character that must not be dialed.
For instance, in a digit map such as “25#”, the “#” signals that the user has finished entering digits. To remove the “#”, specify “#” in this field and the resulting number is “25”.
This rule is applied AFTER applying the *Remove Prefix* (Step 3) rule, but BEFORE applying *Add the Prefix* (Step 4) rule.
6. Specify the line(s) on which to apply the digit map in the *Line To Apply* column.
The string has the following syntax:
 - **all**: Applies to all lines.

- ,: Separator between non-consecutive lists of lines or single line.
- *n*: A single line, where n is the line number.
- *m-n*: List of lines where m is the start line number and n is the end line number.



Note: Line duplication is not allowed. Lines must be specified in low to high order.

Example:

'1,3-6': Applies to lines 1, 3, 4, 5, and 6.

The default value is **all**.

7. Click *Submit* if you do not need to set other parameters.

Blocked Digit Maps

A blocked digit map forbids to call specific numbers; for instance, you want to accept all 1-8xx numbers except 1-801. You can create/edit ten blocked digit maps for the Medatrix 4100.

► **To set up blocked digit maps:**

1. In the *Blocked Digit Map* section – *Activation* column, enable one or more digit maps by selecting the corresponding **Enable** choice.

Figure 40: Blocked Digit Map Section

Blocked Digit Map			
Index	Activation	Digit Map	Line To Apply
1	☐ Enable ☒ Disable		all
2	☐ Enable ☒ Disable		all
3	☐ Enable ☒ Disable		all
4	☐ Enable ☒ Disable		all
5	☐ Enable ☒ Disable		all
6	☐ Enable ☒ Disable		all
7	☐ Enable ☒ Disable		all
8	☐ Enable ☒ Disable		all
9	☐ Enable ☒ Disable		all
10	☐ Enable ☒ Disable		all

2. Define the digit map string that is considered invalid when dialed in the *Digit Map* column. The string must use the syntax described in [“Digit Maps” on page 73](#). The string format is validated upon entry. Invalid entries are refused. A digit map string may have a maximum of 64 characters.
3. Specify the line(s) on which to apply the digit map in the *Line To Apply* column. The string has the following syntax:
- **all**: Applies to all lines.
 - ,: Separator between non-consecutive lists of lines or single line.
 - *n*: A single line, where n is the line number.
 - *m-n*: List of lines where m is the start line number and n is the end line number.



Note: Line duplication is not allowed. Lines must be specified in low to high order.

Example:

'1,3-6': Applies to lines 1, 3, 4, 5, and 6.

The default value is **all**.

4. Click *Submit* if you do not need to set other parameters.

Voice & Fax Codecs

The lines of the Mediatix 4100 can simultaneously use the same codec (for instance, G.711 PCMA), or a mix of any of the supported codecs. Set and enable these codecs for **each** line.

Table 53: Codecs Comparison

	Compression	Voice Quality
G.711	None	Excellent
G.723.1	Highest	Good
G.726	Medium	Fair
G.729a/ab	High	Fair/Good

G.711 PCMA and PCMU

Specified in ITU-T Recommendation G.711. The audio data is encoded as 8 bits per sample, after logarithmic scaling. PCMU denotes μ -law scaling, PCMA A-law scaling.

Table 54: G.711 Features

Feature	Description
Packetization time	Range of 10 ms to 100 ms with increment of 10 ms. See "G.711 Codec Parameters" on page 84 for more details.
Voice Activity Detection (VAD)	Can be enabled or disabled. See "G.711 Codec Parameters" on page 84 for more details.
Comfort noise	Supports comfort noise as defined in <i>draft-ietf-avt-rtp-cn-05.txt</i> . See "G.711 Codec Parameters" on page 84 for more details.

Analog Modem

The Mediatix 4100 can send modem transmissions in clear channel (G.711). If configured adequately, modems with higher rate capabilities (for instance, V.90) will automatically fall back in the transmission range supported.

Quality of modem transmissions is dependent upon the system configuration, quality of the analog lines, as well as the number of analog-to-digital and digital-to-analog conversions. Modem performance may therefore be reduced below the optimum values stated above.

G.726

Specified in ITU-T Recommendation G.726: 40, 32, 24, 16 kbit/s adaptive differential pulse code modulation (ADPCM). It describes the algorithm recommended for conversion of a single 64 kbit/s A-law or U-law PCM channel encoded at 8000 samples/sec to and from a 40, 32, 24, or 16 kbit/s channel. The conversion is applied to the PCM stream using an Adaptive Differential Pulse Code Modulation (ADPCM) transcoding technique.

Table 55: G.726 Features

Feature	Description
Packetization time	Range of 10 ms to 100 ms with increment of 10 ms. See “G.726 Codecs Parameters” on page 86 for more details.
Voice Activity Detection (VAD)	Can be enabled or disabled. See “G.726 Codecs Parameters” on page 86 for more details.
Comfort noise	Supports comfort noise as defined in <i>draft-ietf-avt-rtp-cn-05.txt</i> . See “G.726 Codecs Parameters” on page 86 for more details.

Analog Modem

The Mediatix 4100 can send modem transmissions in clear channel (G.726). If configured adequately, modems with higher rate capabilities (for instance, V.90) will automatically fall back in the transmission range supported.

Quality of modem transmissions is dependent upon the system configuration, quality of the analog lines, as well as the number of analog-to-digital and digital-to-analog conversions. Modem performance may therefore be reduced below the optimum values stated above.

G.723.1

Specified in ITU-T Recommendation G.723.1, dual-rate speech coder for multimedia communications transmitting at 5.3 kbit/s and 6.3 kbit/s. This Recommendation specifies a coded representation that can be used to compress the speech signal component of multi-media services at a very low bit rate. The audio is encoded in 30 ms frames.

A G.723.1 frame can be one of three sizes: 24 octets (6.3 kb/s frame), 20 octets (5.3 kb/s frame), or 4 octets. These 4-octet frames are called SID frames (Silence Insertion Descriptor) and are used to specify comfort noise parameters.

Table 56: G.723.1 Features

Feature	Description
Packetization time	Range of 30 ms to 120 ms with increment of 30 ms. See “G.723 Codec Parameters” on page 85 for more details.
Voice Activity Detection (VAD)	The Mediatix 4100 supports the annex A. Annex A is the built-in support of VAD in G.723.1.

G.729

Specified in ITU-T Recommendation G.729, coding of speech at 8 kbit/s using conjugate structure-algebraic code excited linear prediction (CS-ACELP). For all data rates, the sampling frequency (and RTP timestamp clock rate) is 8000 Hz.

A voice activity detector (VAD) and comfort noise generator (CNG) algorithm in Annex B of G.729 is recommended for digital simultaneous voice and data applications; they can be used in conjunction with G.729 or G.729 Annex A. A G.729 or G.729 Annex A frame contains 10 octets, while the G.729 Annex B comfort noise frame occupies 2 octets.

The Mediatrix 4100 supports G.729A and G.729AB for encoding and G.729, G.729A and G.729AB for decoding.

Table 57: G.729 Features

Feature	Description
Packetization time	Range of 10 ms to 100 ms with increment of 10 ms. See “G.729 Codec Parameters” on page 85 for more details.
Voice Activity Detection (VAD)	The Mediatrix 4100 supports the annex B. Annex B is the built-in support of VAD in G.729. See “G.729 Codec Parameters” on page 85 for more details.

General Parameters

The following are the general codecs parameters you can set.

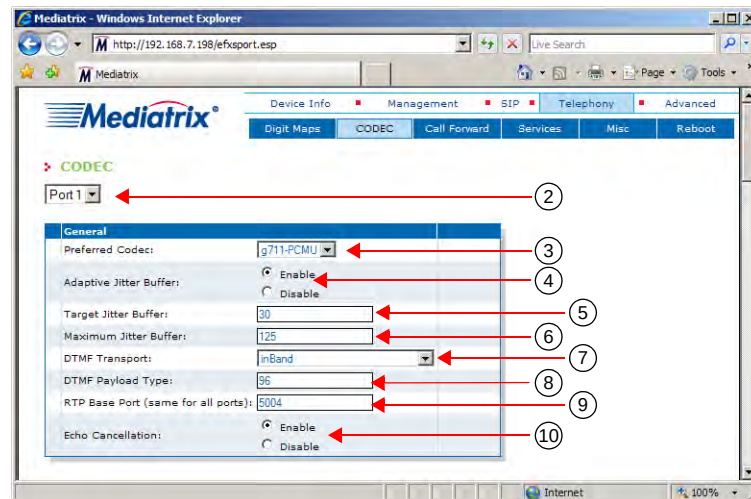
Standards Supported	• draft-choudhuri-sip-info-digit-00.txt • ITU-T Recommendation Q.24 : Multifrequency push-button signal reception • RFC 2833: RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals • RFC 1890 – RTP Profile for Audio and Video Conferences with Minimal Control
----------------------------	---

You can also set these parameters via SNMP, as described in [“Chapter 15 - Voice Transmissions” on page 231](#).

► To set the general codecs parameters:

1. In the web interface, click the *Telephony* link, then the *CODEC* sub-link.

Figure 41: Telephony – CODEC Web Page



2. Select to which port you want to apply the changes in the drop down menu at the top of the window.
3. In the *General* section, choose the preferred codec you want to use in the *Preferred Codec* field. This is the codec you want to favour during negotiation. You have the following choices:
 - g711-PCMU
 - g711-PCMA
 - g723

- g729
- g726-16kbps
- g726-24kbps
- g726-32kbps
- g726-40kbps

The default value is **pcmu**.

4. Enable the jitter buffer protection by selecting **Enable** in the *Adaptive Jitter Buffer* choice.

The jitter buffer allows better protection against packet loss, but increases the voice delay. If the network to which the Mediatrix 4100 is connected suffers from a high level of congestion, the jitter buffer protection level should be higher. If the network to which the Mediatrix 4100 is connected suffers from a low level of congestion, the jitter buffer protection level should be lower.



Note: Do not put **0** as values for the *Target Jitter Buffer* and *Maximum Jitter Buffer* fields.

5. Define the target jitter buffer length in the *Target Jitter Buffer* field.

The adaptive jitter buffer attempts to hold packets to the target holding time. This is the minimum delay the jitter buffer adds to the system. The target jitter buffer length is in ms and must be equal to or smaller than the maximum jitter buffer.

Values range from 0 ms to 135 ms. The default value is 30 ms. You can change values by increments of 1 ms, but Media5 recommends to use multiple of 5 ms.

It is best not to set target jitter values below the default value. Setting a target jitter buffer below 5 ms could cause an error. Jitter buffer adaptation behaviour varies from one codec to another. See [“About Changing Jitter Buffer Values” on page 83](#) for more details.

6. Define the maximum jitter buffer length in the *Maximum Jitter Buffer* field.

This is the maximum jitter the adaptive jitter buffer can handle. The jitter buffer length is in ms and must be equal to or greater than the target jitter buffer.

Values range from 0 ms to 135 ms. The default value is 125 ms. You can change values by increments of 1 ms, but Media5 recommends to use multiple of 5 ms.

The maximum jitter buffer value should be equal to the minimum jitter buffer value + 4 times the ptime value. Let's say for instance that:

- Minimum jitter buffer value is 30 ms
- Ptime value is 20 ms

The maximum jitter buffer value should be: $30 + 4 \times 20 = 110$ ms

See [“About Changing Jitter Buffer Values” on page 83](#) for more details.

7. Set the DTMF transport type in the *DTMF Transport* field.

Table 58: DTMF Transport Type Parameters

Transport Parameter	Description
inBand	The DTMFs are transmitted like the voice in the RTP stream.

DTMF out-of-band

Certain compression codecs such as G.723.1 and G.729 effectively distort voice because they lose information from the incoming voice stream during the compression and decompression phases. For normal speech this is insignificant and becomes unimportant. In the case of pure tones (such as DTMF) this distortion means the receiver may no longer recognize the tones. The solution is to send this information as a separate packet to the other endpoint, which then plays the DTMF sequence back by re-generating the true tones. Such a mechanism is known as out-of-band DTMF. The Mediatrix 4100 receives and sends out-of-band DTMFs as per ITU Q.24. DTMFs supported are 0-9, A-D, *, #.

Table 58: DTMF Transport Type Parameters (Continued)

Transport Parameter	Description
outOfBandUsingRtp	The DTMFs are transmitted as per RFC 2833.
outOfBandUsingSignalingProtocol	The DTMFs are transmitted as per <i>draft-choudhuri-sip-info-digit-00.txt</i> . Note: This feature and the Hook Flash processing feature via signalling protocol are totally independent. Activating one of these features has no effect on the other. See “Hook Flash Processing” on page 324 for more details.
signalingProtocolDependent	The signalling protocol has the control to select the DTMF transport mode. The SDP body includes both RFC 2833 and <i>draft-choudhuri-sip-info-digit-00.txt</i> in that order of preference.

8. Set the payload type in the *DTMF Payload Type* field.

You can determine the actual RTP dynamic payload type used for the “telephone-event” in an initial offer. The payload types available are as per RFC 1890. Available values range from 96 to 127.



Note: This parameter applies only when selecting the *outOfBandUsingRtp* DTMF transport mode.

9. Set the *RTP Base Port* field with the port number you want to use as RTP/RTCP base port. The RTP/RTCP ports are allocated starting from the base port. The Mediatrix 4100 may use two or four RTP/RTCP ports for each FXS interface:
- It uses two ports in case of a standard call.
 - It uses four ports in other types of calls such as a conference call, a call transfer, etc.
- The default RTP/RTCP base port is **5004**. In the case of the base port defined on 5004:
- If there is currently no ongoing call and FXS connector 1 has an incoming or outgoing call, it uses the RTP/RTCP ports 5004 and 5005.
 - If there is currently a standard call on FXS connector 1 and FXS connector 2 has a conference call, then FXS connector 2 uses the RTP/RTCP ports 5006, 5007, 5008, and 5009, which are the next available ports.
10. Select whether the echo cancellation should be enabled or disabled in the *Echo Cancellation* choice.

Table 59: Echo Cancellation Parameters

Parameter	Description
disable	The DSP does not use echo cancellation on the related port.
enable	The DSP proceeds to cancel signals that are recognized as echo when appropriate. This is the default value.

Turning off the echo cancellation feature may be useful to ensure the success of some modem transmissions.

11. Click *Submit* if you do not need to set other parameters.

About Changing Jitter Buffer Values

Media5 recommends to avoid changing the target and maximum jitter buffer values unless experiencing or strongly expecting one of the following symptoms:

- ▶ If the voice is scattered, try to increase the maximum jitter buffer value.
- ▶ If the delay in the voice path (end to end) is too long, you can lower the target jitter value, but **ONLY** if the end-to-end delay measured matches the target jitter value.

For instance, if the target jitter value is 50 ms, the maximum jitter is 135 ms and the delay measured is 130 ms, it would serve nothing to reduce the target jitter. However, if the target jitter value is 100 ms and the measured delay is between 100 ms and 110 ms, then you can lower the target jitter from 100 ms to 30 ms.

G.711 Codec Parameters

The following are the G.711 codec parameters you can set.

► To set the G.711 codec parameters:

1. Select to which port you want to apply the changes in the drop down menu at the top of the window.
2. In the *Voice - G.711* section, enable the G.711 u-Law codec by selecting **Enable** in the *G.711 u-Law* choice.

Figure 42: Telephony – G.711 Section

The screenshot shows the 'Voice - G.711' configuration window. It contains several settings with red arrows and numbered circles (2-7) pointing to specific elements:

- 2: Points to the 'G.711 u-Law' radio button, which is currently set to 'Enable'.
- 3: Points to the 'G.711 u-Law minimum packetization time' dropdown menu, which is set to '10 ms'.
- 4: Points to the 'G.711 a-Law' radio button, which is currently set to 'Disable'.
- 5: Points to the 'G.711 a-Law minimum packetization time' dropdown menu, which is set to '10 ms'.
- 6: Points to the 'G.711 VAD' dropdown menu, which is set to 'conservative'.
- 7: Points to the 'G.711 Comfort Noise Generation' dropdown menu, which is set to 'customNoise'.

3. Set the minimum and maximum packetization time values for the G.711 u-Law codec in the corresponding drop-down menu.
The packetization time (also called packetization period or ptime) is the duration, in ms, of the voice packet.
 - **Minimum:** Shortest packetization period allowed for the G.711 u-Law codec. Authorized values start at 10 ms and come in discrete steps of 10 ms up to the one specified by the corresponding *Maximum* drop-down menu.
 - **Maximum:** Longest packetization period allowed for the G.711 u-Law codec. Authorized values go up to 100 ms, in discrete steps of 10 ms, and start at the one specified by the corresponding *Minimum* drop-down menu.
4. Enable the G.711 a-Law codec by selecting **Enable** in the *G.711 a-Law* field.
5. Set the minimum and maximum packetization time values for the G.711 a-Law codec in the corresponding drop-down menu.
The packetization time (also called packetization period or ptime) is the duration, in ms, of the voice packet.
 - **Minimum:** Shortest packetization period allowed for the G.711 a-Law codec. Authorized values start at 10 ms and come in discrete steps of 10 ms up to the one specified by the corresponding *Maximum* drop-down menu.
 - **Maximum:** Longest packetization period allowed for the G.711 a-Law codec. Authorized values go up to 100 ms, in discrete steps of 10 ms, and start at the one specified by the corresponding *Minimum* drop-down menu.
6. Enable the G.711 Voice Activity Detection (VAD) by selecting **Enable** in the *G.711 VAD* choice.
VAD defines how the Mediatrix 4100 sends information pertaining to silence. This allows the unit to detect when the user talks, thus avoiding to send silent RTP packets. This saves on network resources. However, VAD may affect packets that are not really silent (for instance, cut sounds that are too low). VAD can thus slightly affect the voice quality.

7. Enable the G.711 Comfort Noise Generation (CNG) by selecting **Enable** in the *G.711 Comfort Noise Generation* choice.
Comfort Noise (CN) defines how the Mediatrrix 4100 processes silence periods information it receives. During silence periods, the Mediatrrix 4100 may receive CN packets containing information about background noise. Those packets are used to generate local comfort noise.
8. Click *Submit* if you do not need to set other parameters.

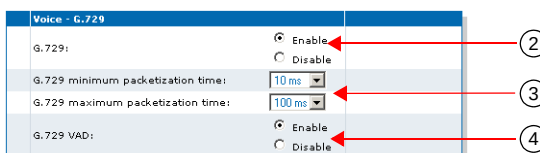
G.729 Codec Parameters

The following are the G.729 codec parameters you can set.

► To set the G.729 codec parameters:

1. Select to which port you want to apply the changes in the drop down menu at the top of the window.
2. In the *Voice - G.729* section, enable the G.729 codec by selecting **Enable** in the *G.729* choice.

Figure 43: Telephony – G.729 Section



3. Set the minimum and maximum packetization time values for the G.729 codec in the corresponding drop-down menu.

The packetization time (also called packetization period or ptime) is the duration, in ms, of the voice packet.

- **Minimum:** Shortest packetization period allowed for the G.729 codec. Authorized values start at 10 ms and come in discrete steps of 10 ms up to the one specified by the corresponding *Maximum* drop-down menu.
- **Maximum:** Longest packetization period allowed for the G.729 codec. Authorized values go up to 100 ms, in discrete steps of 10 ms, and start at the one specified by the corresponding *Minimum* drop-down menu.

4. Enable the G.729 Voice Activity Detection (VAD) by selecting **Enable** in the *G.729 VAD* choice.
VAD defines how the Mediatrrix 4100 sends information pertaining to silence. This allows the unit to detect when the user talks, thus avoiding to send silent RTP packets. This saves on network resources. However, VAD may affect packets that are not really silent (for instance, cut sounds that are too low). VAD can thus slightly affect the voice quality.
G.729 has a built-in VAD in its Annex B version. It is recommended for digital simultaneous voice and data applications and can be used in conjunction with G.729 or G.729 Annex A. A G.729 or G.729 Annex A frame contains 10 octets, while the G.729 Annex B frame occupies 2 octets. The CN packets are sent in accordance with annex B of G.729.
5. Click *Submit* if you do not need to set other parameters.

G.723 Codec Parameters

The following are the G.723 codec parameters you can set.

► To set the G.723 codec parameters:

1. Select to which port you want to apply the changes in the drop down menu at the top of the window.
2. In the *Voice - G.723* section, enable the G.723 codec by selecting the proper value in the *G.723* drop-down menu.

Figure 44: Telephony – G.723 Section

Voice - G.723	
G.723:	g723-63kbps
G.723 minimum packetization time:	30 ms
G.723 maximum packetization time:	120 ms

- Set the minimum and maximum packetization time values for the G.723 codec in the corresponding drop-down menu.
The packetization time (also called packetization period or ptime) is the duration, in ms, of the voice packet.
 - Minimum:** Shortest packetization period allowed for the G.723 codec. Authorized values start at 30 ms and come in discrete steps of 30 ms up to the one specified by the corresponding **Maximum** drop-down menu.
 - Maximum:** Longest packetization period allowed for the G.723 codec. Authorized values go up to 120 ms, in discrete steps of 30 ms, and start at the one specified by the corresponding **Minimum** drop-down menu.
- Click *Submit* if you do not need to set other parameters.

G.726 Codecs Parameters

The following are the G.726 codecs parameters you can set. There are sections for each type of G.726 codec.

► To set the G.726 codecs parameters:

- Select to which port you want to apply the changes in the drop down menu at the top of the window.
- In one or more of the *Voice - G.726* sections, enable the corresponding G.726 codec by selecting **Enable** in the *G.726* choice.

Figure 45: Telephony – G.726 Section

Voice - G.726 16Kbps	
G.726 16Kbps:	☐ Enable ☒ Disable
G.726 16Kbps Payload Type:	97
G.726 16Kbps Minimum Packetization Time:	10 ms
G.726 16Kbps Maximum Packetization Time:	100 ms

- Set the G.726 actual RTP dynamic payload type used in an initial offer in the *G.726 Payload Type* field.
The payload types available are as per RFC 3551. The values range from 96 to 127. The default values are as follows:
 - G.726 16 kbps: 97.
 - G.726 24 kbps: 98.
 - G.726 32 kbps: 99.
 - G.726 40 kbps: 100.



Note: When selecting the dynamic payload type, make sure that the value is not already used by another dynamic codec. If a value between 96 and 127 is refused, this means it is already used by another dynamic codec.



Note: If you set the *DTMF Transport* field to **outOfBandUsingSignalingProtocol** ("[General Parameters](#)" [on page 81](#)), you cannot configure a dynamic payload type to 111 because it is already used by the DTMF out-of-band using signalling protocol.

4. Set the minimum and maximum packetization time values for the G.726 codec in the corresponding drop-down menu.
The packetization time (also called packetization period or ptime) is the duration, in ms, of the voice packet.
 - **Minimum:** Shortest packetization period allowed for the G.726 codec. Authorized values start at 10 ms and come in discrete steps of 10 ms up to the one specified by the corresponding *Maximum* drop-down menu.
 - **Maximum:** Longest packetization period allowed for the G.726 codec. Authorized values go up to 100 ms, in discrete steps of 10 ms, and start at the one specified by the corresponding *Minimum* drop-down menu.
5. Click *Submit* if you do not need to set other parameters.

Fax Parameters

The Mediatrix 4100 handles G3 fax transmissions at speeds up to 14.4 kbps. Automatic fax mode detection is standard on all lines. Real-Time Fax Over UDP with the T.38 protocol stack is also available.

The quality of T.38 fax transmissions depends upon the system configuration, type of call control system used, type of Mediatrix units deployed, as well as the model of fax machines used. Should some of these conditions be unsatisfactory, performance of T.38 fax transmissions may vary and be reduced below expectations.

A fax call works much like a regular voice call, with the following differences:

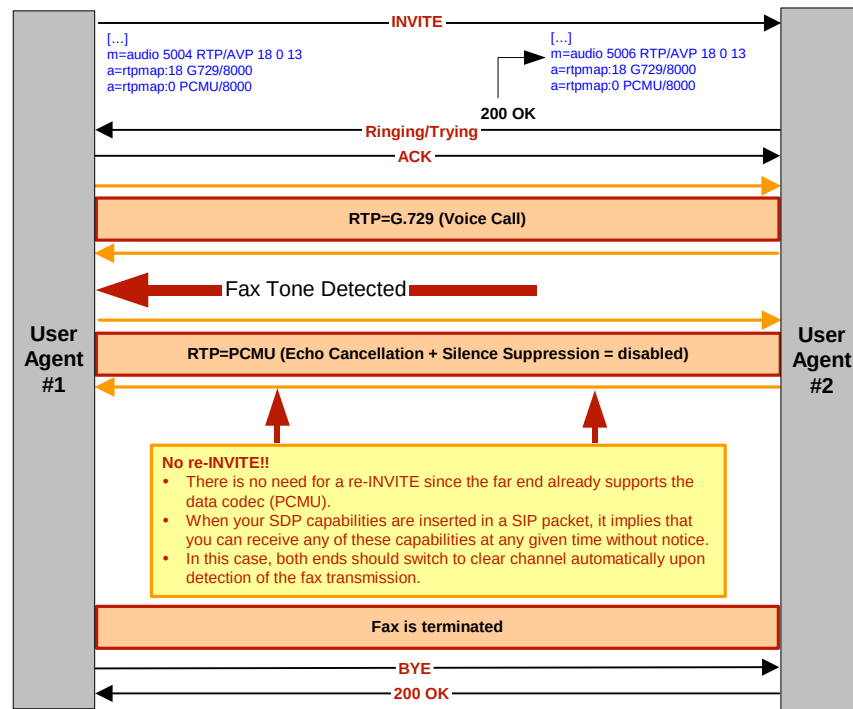
1. The fax codec may be re-negotiated by using a re-INVITE.
2. The goal of the re-INVITE is to allow both user agents to agree on a fax codec, which is either:
 - a. Clear channel (PCMU/PCMA or G.726) without Echo Cancellation nor Silence Suppression (automatically disabled).
 - b. T.38.
3. Upon fax termination, if the call is not BYE, the previous voice codec is recovered with another re-INVITE.

All lines of the Mediatrix 4100 can simultaneously use the same codec (for instance, T.38), or a mix of any of the supported codecs. Set and enable these codecs for **each** line.

Clear Channel Fax

The Mediatrix 4100 can send faxes in clear channel. The following is a clear channel fax call flow:

Figure 46: Clear Channel Fax Call Flow



T.38 Fax

Standards Supported

- Based on *draft-ietf-sipping-realtimefax-01.txt*
- Recommendation ITU T.38 version 0

T.38 fax relay is a real-time fax transmission; that is, two fax machines communicating with each other as if there were a direct phone line between the two. T.38 is called a fax relay, which means that instead of sending inband fax signals, which implies a loss of signal quality, it sends those fax signals out-of-band in a T.38 payload, so that the remote end can reproduce the signal locally.

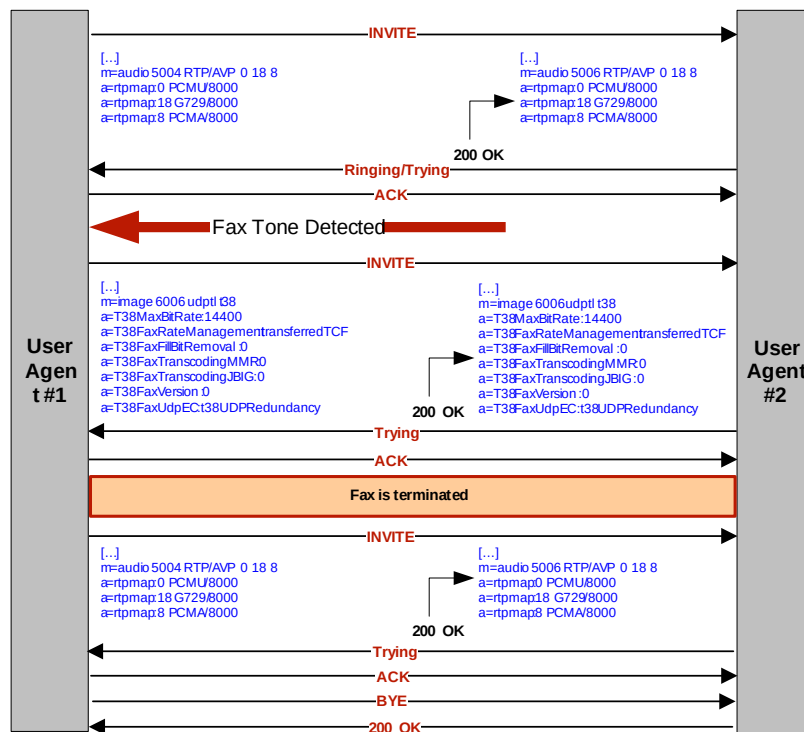
The Mediatrix 4100 can send faxes in T.38 mode over UDP or TCP. T.38 is used for fax if both units are T.38 capable; otherwise, transmission in clear channel over G.711 as defined is used (if G.711 *μ-law* and/or G.711 *A-law* are enabled). If no clear channel codecs are enabled and the other endpoint is not T.38 capable, the fax transmission fails.



Caution: The Mediatrix 4100 opens the T.38 channel only after receiving the “200 OK” message from the peer. This means that the Mediatrix 4100 cannot receive T.38 packets before receiving the “200 OK”. Based on RFC 3264, the T.38 channel should be opened as soon as the unit sends the “INVITE” message. See [“Fax Issues” on page 355](#) for more details.

The following is a T.38 fax call flow:

Figure 47: T.38 Fax Call Flow



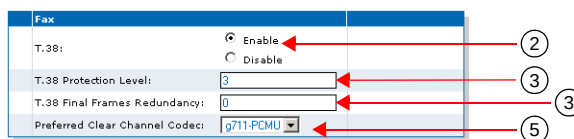
Fax Parameters Configuration

The following are the fax codecs parameters you can set. You can also set these parameters via SNMP, as described in [“Chapter 16 - Fax Transmission” on page 249](#).

► To set the fax codecs parameters:

1. Select to which port you want to apply the changes in the drop down menu at the top of the window.
2. In the *Fax* section, enable the T.38 codec by selecting **Enable** in the *T.38* choice.

Figure 48: Telephony – Fax Section



3. Set the number of redundancy packets sent with the current packet in the *T.38 Protection Level* field.
This is the standard redundancy offered by T.38. Please see step 4 for additional reliability options for T.38. Available values range from 1 to 5.
4. For additional reliability, define the number of times T.38 packets are retransmitted in the *T.38 Final Frames Redundancy* field.
This only applies to the T.38 packets where the PrimaryUDPTL contains the following T.38 data type:
 - HDLC_SIG_END,
 - HDLC_FCS_OK_SIG_END,
 - HDLC_FCS_BAD_SIG_END and
 - T4_NON_ECM_SIG_END

5. Set the clear channel codec to use upon detecting a fax tone in the *Preferred Clear Channel Codec* field.

This is used to decide which of the following codecs is preferred, even for voice transmissions:

- PCMU
- PCMA
- G.726 at 32 kbs
- G.726 at 40 kbs



Note: In clear channel, G.726 at 16 kbs and 24 kbs are not recommended for fax transmission.

- noPreferredCodec

When *noPreferredCodec* is selected and no data-capable codecs are negotiated, data transmission may fail.

This parameter increases the relative priority of the selected codec vs other data-capable codecs. However, the priority of the preferred clear channel codec remains lower than the voice's *Preferred Codec* field. (see [“General Parameters” on page 76](#)).

Moreover, when no data-capable codec is part of the list of negotiated codecs, this variable indicates which codec to use when fax or modem tones are detected. However, if the negotiated voice codec is data-capable, the voice codec will be used for data instead of the preferred data codec. See [“Data Codec Selection Procedure” on page 253](#) for more details.

6. Click *Submit* if you do not need to set other parameters.

Call Forward

The *Call Forward* sub-page of the *Telephony* page allows you to set the three types of Call Forward:

- ▶ On Busy
- ▶ On No Answer
- ▶ Unconditional

You can also set these parameters via SNMP, as described in [“Call Forward” on page 308](#).

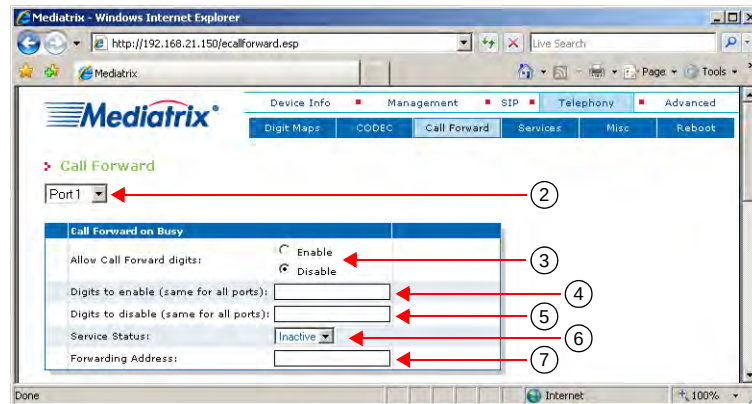
On Busy

You can automatically forward the incoming calls of your users to a pre-determined target if they are already on the line. The user does not have any feedback that a call was forwarded.

► **To set the Call Forward On Busy feature:**

1. In the web interface, click the *Telephony* link, then the *Call Forward* sub-link.

Figure 49: Telephony – Call Forward Web Page



2. Select to which port you want to apply the changes in the drop down menu at the top of the window.
3. In the *Call Forward on Busy* section, enable the service by selecting **Enable** in the *Allow Call Forward digits* choice.
If you select **Disable**, this will not disable the call forward, but will prevent the user from activating or deactivating the service. The user will not be able to use the digits used to start and stop the service.
4. Define the digits that users must dial to start the service in the *Digits to enable* field.
Define this field only if you want to let the user start the service with his or her handset. If you rather want to have the control over the activation/deactivation of the service, proceed to Step 6.
For instance, you could decide to put “*72” as the sequence to activate the service. This sequence must be unique and follow the syntax for digit maps (see [“Digit Maps” on page 73](#)). Dialing this digit map does not have any effect unless the service’s status is “enabled”.
The activating sequence is set for all the lines of the Mediatix 4100. You cannot have a different sequence for each line.
5. Define the digits that users must dial to stop the service in the *Digits to disable* field.
Define this field only if you want to let the user stop the service with his or her handset. If you rather want to have the control over the activation/deactivation of the service, proceed to Step 6.
For instance, you could decide to put “*73” as the sequence to deactivate the service. This sequence must be unique and follow the syntax for digit maps (see [“Digit Maps” on page 73](#)). Dialing this digit map does not have any effect unless the service’s status is “enabled”.
The deactivating sequence is set for all the lines of the Mediatix 4100. You cannot have a different sequence for each line.
6. Set the activation status of the service in the *Service Status* field to **Inactive** or **Active**.
This feature starts the service (active) or stops the service (inactive).
If you want to let the user start or stop the service with his or her handset, you must enter a sequence of digits in steps 4 and 5. The *Service Status* field is automatically updated to reflect the activation status according to the user’s setting.
7. Define the address to which forward incoming calls in the *Forwarding Address* field.
Accepted formats are:
 - telephone numbers (5551111)

- SIP URLs such as "scheme:user@host". For instance, "sip:user@foo.com".

This string is used literally, so cosmetic symbols (such as the dash in "555-xxxx") should not be present.

8. Click *Submit* if you do not need to set other parameters.

Using Call Forward on Busy

The following is the procedure to use this service on the user's telephone.

► To forward calls:

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence implemented to activate the call forward on busy service.
This sequence could be something like *72.
4. Wait for the transfer tone (three "beeps") followed by the dial tone.
5. Dial the number to which you want to forward your calls. Dial any access code if required.
6. Wait for three "beeps" followed by a silent pause.
The call forward is established.
7. Hang up your telephone.

► To cancel the call forward:

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence implemented to deactivate the call forward on busy service.
This sequence could be something like *73.
4. Wait for three "beeps" followed by a silent pause.
The call forward is cancelled.
5. Hang up your telephone.

On No Answer

You can forward the incoming calls of your users to a pre-determined target if they do not answer their telephone before a specific amount of time. The user does not have any feedback that a call was forwarded.

► **To set the Call Forward On No Answer feature:**

1. Select to which port you want to apply the changes in the drop down menu at the top of the window.
2. In the *Call Forward on No Answer* section, enable the service by selecting **Enable** in the *Allow Call Forward digits* choice.

Figure 50: Telephony – Call Forward on No Answer section

If you select **Disable**, this will not disable the call forward, but will prevent the user from activating or deactivating the service. The user will not be able to use the digits used to start and stop the service.

3. Define the digits that users must dial to start the service in the *Digits to enable* field.
Define this field only if you want to let the user start the service with his or her handset. If you rather want to have the control over the activation/deactivation of the service, proceed to Step 6.
For instance, you could decide to put “*74” as the sequence to activate the service. This sequence must be unique and follow the syntax for digit maps (see [“Digit Maps” on page 73](#)). Dialing this digit map does not have any effect unless the service’s status is “enabled”.
The activating sequence is set for all the lines of the Mediatrix 4100. You cannot have a different sequence for each line.
4. Define the digits that users must dial to stop the service in the *Digits to disable* field.
Define this field only if you want to let the user stop the service with his or her handset. If you rather want to have the control over the activation/deactivation of the service, proceed to Step 6.
For instance, you could decide to put “*75” as the sequence to deactivate the service. This sequence must be unique and follow the syntax for digit maps (see [“Digit Maps” on page 73](#)). Dialing this digit map does not have any effect unless the service’s status is “enabled”.
The deactivating sequence is set for all the lines of the Mediatrix 4100. You cannot have a different sequence for each line.
5. Define the time, in milliseconds, the telephone keeps ringing before the call forwarding activates in the *Timeout Value* field.
6. Set the status of the service in the *Service Status* field to **Inactive** or **Active**.
This feature starts the service (active) or stops the service (inactive).
If you want to let the user start or stop the service with his or her handset, you must enter a sequence of digits in steps 3 and 4. The *Service Status* field is automatically updated to reflect the activation status according to the user’s setting.
7. Define the address to which forward incoming calls in the *Forwarding Address* field.
Accepted formats are:
 - telephone numbers (5551111)
 - SIP URLs such as “scheme:user@host”. For instance, “sip:user@foo.com”.
 This string is used literally, so cosmetic symbols (such as the dash in “555-xxxx”) should not be present.
8. Click *Submit* if you do not need to set other parameters.

Using Call Forward on No Answer

The following is the procedure to use this service on the user's telephone.

► To forward calls:

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence implemented to activate the call forward on no answer service.
This sequence could be something like *74.
4. Wait for the transfer tone (three “beeps”) followed by the dial tone.
5. Dial the number to which you want to forward your calls. Dial any access code if required.
6. Wait for three “beeps” followed by a silent pause.
The call forward is established.
7. Hang up your telephone.

► To cancel the call forward:

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence implemented to deactivate the call forward on no answer service.
This sequence could be something like *75.
4. Wait for three “beeps” followed by a silent pause.
The call forward is cancelled.
5. Hang up your telephone.

Unconditional

The Call Forward Unconditional feature allows users to forward all of their calls to another extension or line.

► To set the Call Forward Unconditional feature:

1. Select to which port you want to apply the changes in the drop down menu at the top of the window.
2. In the *Call Forward Unconditional* section, enable the service by selecting **Enable** in the *Allow Call Forward digits* choice.

Figure 51: Telephony – Call Forward Unconditional Section

Call Forward Unconditional	
Allow Call Forward digits:	☐ Enable ☒ Disable
Digits to enable (same for all ports):	
Digits to disable (same for all ports):	
Service Status:	Inactive
Forwarding Address:	

If you select **Disable**, this will not disable the call forward, but will prevent the user from activating or deactivating the service. The user will not be able to use the digits used to start and stop the service.

3. Define the digits that users must dial to start the service in the *Digits to enable* field.
Define this field only if you want to let the user start the service with his or her handset. If you rather want to have the control over the activation/deactivation of the service, proceed to Step 5.
For instance, you could decide to put “*74” as the sequence to activate the service. This sequence must be unique and follow the syntax for digit maps (see [“Digit Maps” on page 73](#)). Dialing this digit map does not have any effect unless the service's status is “enabled”.

The activating sequence is set for all the lines of the Mediatrix 4100. You cannot have a different sequence for each line.

4. Define the digits that users must dial to stop the service in the *Digits to disable* field.
Define this field only if you want to let the user stop the service with his or her handset. If you rather want to have the control over the activation/deactivation of the service, proceed to Step 5.
For instance, you could decide to put “*75” as the sequence to deactivate the service. This sequence must be unique and follow the syntax for digit maps (see [“Digit Maps” on page 73](#)). Dialing this digit map does not have any effect unless the service’s status is “enabled”.
The deactivating sequence is set for all the lines of the Mediatrix 4100. You cannot have a different sequence for each line.
5. Set the status of the service in the *Service Status* field to **Inactive** or **Active**.
This feature starts the service (active) or stops the service (inactive).
If you want to let the user start or stop the service with his or her handset, you must enter a sequence of digits in steps 3 and 4. The *Service Status* field is automatically updated to reflect the activation status according to the user’s setting.
6. Define the address to which forward incoming calls in the *Forwarding Address* field.
Accepted formats are:
 - telephone numbers (5551111)
 - SIP URLs such as “scheme:user@host”. For instance, “sip:user@foo.com”.This string is used literally, so cosmetic symbols (such as the dash in “555-xxxx”) should not be present.
7. Click *Submit* if you do not need to set other parameters.

Using Call Forward Unconditional

When forwarding calls outside the system, a brief ring is heard on the telephone to remind the user that the call forward service is active. The user can still make calls from the telephone.

► To forward calls:

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence implemented to activate the call forward unconditional service.
This sequence could be something like *70.
4. Wait for the transfer tone (three “beeps”) followed by the dial tone.
5. Dial the number to which you want to forward your calls. Dial any access code if required.
6. Wait for three “beeps” followed by a silent pause.
The call forward is established.
7. Hang up your telephone.

► To check if the call forward has been properly established:

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial your extension or telephone number.
The call is forwarded to the desired telephone number.
4. Hang up your telephone.

► To cancel the call forward:

1. Take the receiver off-hook.

2. Wait for the dial tone.
3. Dial the sequence implemented to deactivate the call forward – unconditional service. This sequence could be something like *71.
4. Wait for three “beeps” followed by a silent pause. The call forward is cancelled.
5. Hang up your telephone.

Services

The *Services* sub-page of the *Telephony* page allows you to set the following subscriber services:

- ▶ Call Transfer
- ▶ Call Waiting
- ▶ Conference
- ▶ Hold
- ▶ Second call
- ▶ Automatic call

You can also set these parameters via SNMP, as described in [“Chapter 23 - Subscriber Services” on page 305](#).

Call Transfer

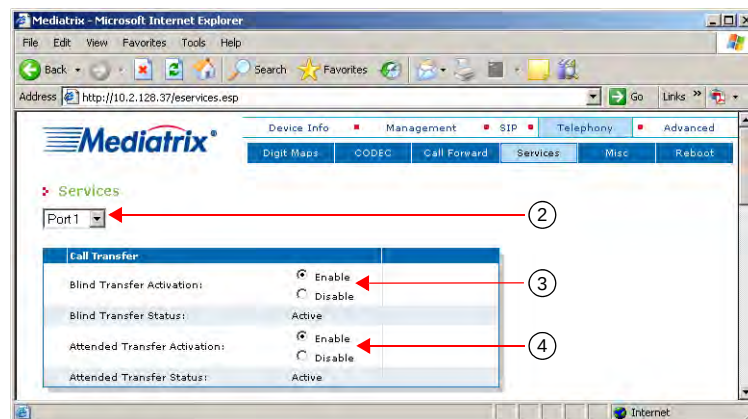
The Call Transfer service offers two ways to transfer calls:

- ▶ Blind Transfer
- ▶ Attended Transfer

▶ To enable the Call Transfer services:

1. In the web interface, click the *Telephony* link, then the *Services* sub-link.

Figure 52: Telephony – Call Transfer Web Page



2. Select to which port you want to apply the changes in the drop down menu at the top of the window.
3. In the *Call Transfer* section, enable the Blind Transfer service by selecting **Enable** in the *Blind Transfer Activation* choice.

The blind call transfer service is sometimes called Transfer without Consultation or Unattended Transfer. It allows a user to transfer a call on hold to a still ringing (unanswered) call. The individual at the other extension or telephone number does not need to answer to complete the transfer.

The call hold and second call services must be enabled for this service to work. See [“Call Hold” on page 102](#) and [“Second Call” on page 103](#).

4. Enable the Attended Transfer service by selecting **Enable** in the *Attended Transfer Activation* choice.

The attended call transfer service is sometimes called Transfer with Consultation. It allows a user to transfer a call on hold to an active call. The individual at the other extension or telephone number must answer to complete the transfer.

The call hold and second call services must be enabled for this service to work. See [“Call Hold” on page 102](#) and [“Second Call” on page 103](#).

5. Click *Submit* if you do not need to set other parameters.

Using Blind Call Transfer

The following is the procedure to use this service on the user's telephone.

► To transfer a current call blind:

1. Perform a Flash-Hook by pressing the “Flash” button on your analog telephone.
This puts the call on hold.
2. Wait for the transfer tone (three “beeps”).
3. Dial the number to which you want to transfer the call.
4. Wait for the ringback tone, then hang up your telephone.
The call is transferred. You can also wait for the third party to answer if you want. In this case, the call transfer becomes attended.
If you want to get back to the first call (the call on hold), you must perform two Flash-Hooks.
You are back with the first call and the third party is released.

Using Attended Call Transfer

The following is the procedure to use this service on the user's telephone.

► To transfer a current call attended:

1. Perform a Flash-Hook by pressing the “Flash” button on your analog telephone.
This puts the call on hold.
2. Wait for the transfer tone (three “beeps”).
3. Dial the number to which you want to transfer the call.
The third party answers.
4. Hang up your telephone.
The call is transferred.
5. If you want to get back to the first call (the call on hold), you must perform two Flash-Hooks.
You are back with the first call and the third party is released.



Note: If the number to which you want to transfer the call is busy or does not answer, quickly perform a Flash-Hook. The busy tone or ring tone is cancelled and you are back with the first call.

Call Waiting

The call waiting tone indicates to an already active call that a new call is waiting on the second line.

Your users can activate/deactivate the call waiting tone for their current call. This is especially useful when transmitting faxes. The user that is about to send a fax can thus deactivate the call waiting tone to ensure that the fax transmission will not be disrupted by an unwanted second call. When the fax transmission is completed and the line is on-hook, the call waiting tone is automatically reactivated.

Your users can also permanently activate/deactivate the call waiting service.

Furthermore, the Mediatrix 4100 supports receiving some Call Waiting control commands via the SIP INFO method. See [“Controlling the Call Waiting Tone via SIP INFO” on page 283](#) for more details.

► **To set the Call Waiting services:**

1. Select to which port you want to apply the changes in the drop down menu at the top of the window.
2. In the *Call Waiting* section, enable the service by selecting **Enable** in the *Activation* choice.

Figure 53: Call Waiting Section

This permanently activates the call waiting tone. When receiving new calls during an already active call, a special tone is heard to indicate that a call is waiting on the second line. The user can then answer that call by using the “flash” button. The user can switch between the two active calls by using the “flash” button.

The call hold service must be enabled for this service to work. See [“Call Hold” on page 102](#).

If the user is exclusively using faxes, select **Disable** to permanently disable the call waiting tone.

The user may cancel this service on a per-call basis when dialing a DTMF sequence matching the digit map stored in the *Cancel Digit Map* field (see Step 3). The user may also disable or enable this service permanently with the *Permanent Activation Digit Map* and *Permanent Deactivation Digit Map* fields (See Step 4).

3. Define the digits that users must dial to disable the Call Waiting tone in the *Per Call Deactivation Digit Map* field.

This allows a user who has call waiting enabled to disable that service on the next call only. If, for any reason, the user wishes to undo the cancel, simply unhook and re-hook the telephone to reset the service.

For instance, you could decide to put “*76” as the sequence to disable the call waiting tone. This sequence must be unique and follow the syntax for digit maps (see [“Digit Maps” on page 73](#)). Dialing this digit map does not have any effect unless the service’s status is “enabled”.

The deactivating sequence is set for all the lines of the Mediatrix 4100. You cannot have a different sequence for each line.

4. Define the digits that users must dial to enable the call waiting service permanently in the *Permanent Activation Digit Map* field.

This activation is permanent until the user deactivates the service as in Step 5.

For instance, you could decide to put “*84” as the sequence to enable the service. This sequence must be unique and follow the syntax for digit maps (see [“Chapter 21 - Digit Maps” on page 293](#)).

The sequence is set for all the lines of the Mediatrix 4100. You cannot have a different sequence for each line.

When dialing this digit map, this sets the *Activation* drop-down menu for the line the user is currently using to **enable**.

5. Define the digits that users must dial to disable the call waiting service permanently in the *Permanent Deactivation Digit Map* field.

This deactivation is permanent until the user enables the service as in Step 4.

For instance, you could decide to put “*85” as the sequence to disable the service. This sequence must be unique and follow the syntax for digit maps (see [“Chapter 21 - Digit Maps” on page 293](#)).

The sequence is set for all the lines of the Mediatrix 4100. You cannot have a different sequence for each line.

When dialing this digit map, this sets the *Activation* drop-down menu for the line the user is currently using to **disable**.

6. Click *Submit* if you do not need to set other parameters.

Using Call Waiting

The call waiting feature alerts the user if he or she is already on the telephone and a second call happens. A “beep” (the call waiting tone) is heard and repeated every ten seconds to indicate there is a second incoming call.

► To put the current call on hold:

1. Perform a Flash-Hook by pressing the “Flash” button on your analog telephone.
This puts the call on hold and the second line is automatically connected to your line.
2. Answer the call on the second line.

► To switch from one line to the other:

1. Perform a Flash-Hook each time you want to switch between lines.

► To terminate the first call before answering the second call:

1. Hang up the telephone.
2. Wait for the telephone to ring.
3. Answer the telephone.
The second call is on the line.

Removing the Call Waiting Tone

You can temporarily activate/deactivate the call waiting tone indicating a call is waiting. This is especially useful when transmitting faxes. If you are about to send a fax, you can thus deactivate the call waiting tone to ensure that the fax transmission is not disrupted by an unwanted second call. When the fax transmission is completed and the line is on-hook, the call waiting tone is automatically reactivated.

► To deactivate the call waiting tone:

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence implemented to deactivate the call waiting tone.
This sequence could be something like *70.
4. Wait for the transfer tone (three “beeps”) followed by the dial tone.
The call waiting tone is disabled.

► To re-enable the call waiting tone:

1. Take the receiver off-hook.
2. Replace the receiver on-hook.
The call waiting tone is re-enabled.

Permanently Removing the Call Waiting Tone

You can permanently activate/deactivate the call waiting service.

► To activate the call waiting service:

1. Take the receiver off-hook.

2. Wait for the dial tone.
3. Dial the sequence the system administrator has implemented to activate the call waiting tone service.
This sequence could be something like *84.
4. Wait for the transfer tone (three “beeps”) followed by the dial tone.
5. Hang up your telephone.
The call waiting tone is enabled.

► **To cancel the call waiting service:**

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence the system administrator has implemented to deactivate the call waiting tone service.
This sequence could be something like *85.
4. Wait for the transfer tone (three “beeps”) followed by the dial tone.
The call waiting is cancelled.
5. Hang up your telephone.

Conference

The Conference Call service allows a user to link two or more calls together to form a single conversation, called a conference.

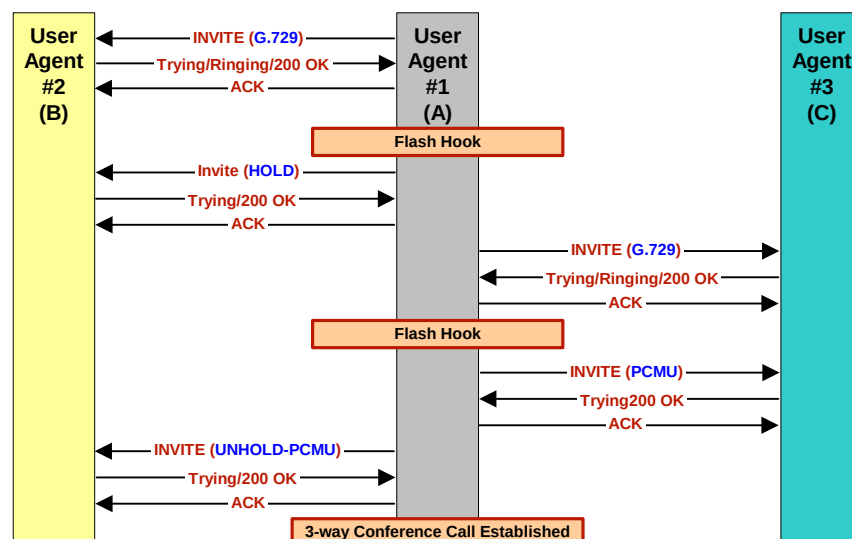
A participant of the conference can put the conference on hold and attempt other calls. This participant may then rejoin the conference at a later time by unholding it. The participant who initiated the conference cannot put it on hold.

You must enable the call hold and second call services for this service to work. See [“Call Hold” on page 102](#) and [“Second Call” on page 103](#).

Furthermore, you must also enable the Attended Transfer service for the two other participants to stay connected once the participant who initiated the conference has hung up. See [“Call Transfer” on page 96](#).

The following is a conference call flow example:

Figure 54: Conference Call Flow



Requirements

For the conference call to occur successfully, all parties must meet the following requirements:

- ▶ Support at least one of the PCM codecs (G.711 μ -law and G.711 A-law) enabled on the line that is having the conference. See [“Voice & Fax Codecs” on page 79](#) for more details.
- ▶ Ability to dynamically change codec during a call.
- ▶ The packetization period (ptime) should be the same for all the participants of the conference. If this is not the case, then part of the conversation may be lost, resulting in a choppy voice. For better results, Media5 recommends to set the packetization period of all participants of a 3-way conference to 30 milliseconds. See [“Voice & Fax Codecs” on page 79](#) for more information on how to set the packetization period of the Mediatix 4100.

Enabling the Conference Call Feature

You must enable this service before your users can use it.

▶ To enable the Conference service:

1. Select to which port you want to apply the changes in the drop down menu at the top of the window.
2. In the *Conference* section, enable the service by selecting **Enable** in the *Activation* choice.

Figure 55: Conference Section

3. Specify how to manage a SIP conference in the *Conference Type* drop-down menu. This configuration only applies to a conference initiated by one of the unit's endpoint.

Table 60: Conference Type Parameters

Parameter	Description
Local	The conference is managed locally by the unit. The maximum number of participants is 3. This is the default value.
Conference Server	The conference is managed by a remote SIP conference server. When using this conference type, both the initiator and a participant of the conference can add new participants to the conference.

4. If you have selected **Conference Server** in the previous step, set the *Conference Address* field with the address of a conference server.
The format must be a SIP URI such as "scheme:user@host". For instance, "sip:user@foo.com".
5. Click *Submit* if you do not need to set other parameters.

Managing a Conference Call

If you are on the telephone with one person and want to conference with a third one, you can do so. In the following examples, let's assume that:

- ▶ "A" is the conference initiator.
- ▶ "B" is the person called on the first line.
- ▶ "C" is the person called on the second line.
- ▶ "D" is a fourth person that "A" wants to add to the conference in **Conference Server** conference type.

► **To initiate a conference (“A” and “B” already connected):**

1. “A” performs a Flash-Hook.
This puts “B” on hold and the second line is automatically connected. “A” hears a dial tone.
2. “A” dials “C’s” number.
“A” and “C” are now connected.
3. “A” performs another Flash-Hook.
The call on hold (“B”) is reactivated. “A” is now conferencing with “B” and “C”.

► **“A” wants to transfer “B” to “C” during the conference:**

This is available only in the **Local** conference type.

1. “A” hangs up.
The conference is terminated. “B” and “C” are now connected.

► **“A” wants to terminate the call with “C” and get back to the call with “B” during the conference:**

This is available only in the **Local** conference type.

1. “A” performs a Flash-Hook.
The conference is terminated and the call with “C” is disconnected. “A” and “B” are still connected and can go on with their conversation.

► **“B” (or “C”) hangs up during the conference:**

This is available only in the **Local** conference type.

1. “B” (or “C”) hangs up during the conference.
The conference is terminated, but the call between “A” and “C” (or “B”) is not affected and they are still connected.

► **“A” wants to add a fourth member to the conference:**

This is available only in the **Converence Server** conference type.

1. “A” performs a Flash-Hook.
This puts “B” and “C” on hold and the second line is automatically connected. “A” hears a dial tone.
2. “A” dials “D’s” number.
“A” and “D” are now connected.
3. “A” performs another Flash-Hook.
The call on hold (“B” and “C”) is reactivated. “A” is now conferencing with “B”, “C”, and “D”.

Call Hold

The Call Hold service allows the user to temporarily put an existing call on hold, usually by using the “flash” button of the telephone. The user can resume the call in the same way.

You must enable this service for the following services to work properly:

- Call Waiting
- Second Call
- Blind Transfer
- Attended Transfer
- Conference

► **To enable the Call Hold service:**

1. Select to which port you want to apply the changes in the drop down menu at the top of the window.
2. In the *Hold* section, enable the service by selecting **Enable** in the *Activation* choice.

Figure 56: Hold Section

Hold	
Activation:	☒ Enable ☐ Disable
Status:	Active

3. Click *Submit* if you do not need to set other parameters.

Using Call Hold

The following is the procedure to use this service on the user's telephone.

► To put the current call on hold:

1. Perform a Flash-Hook by pressing the “Flash” button on your analog telephone.
This puts the call on hold. You can resume the call in the same way.

Second Call

The Second Call service allows a user with an active call to put the call on hold, and then initiate a new call on a second line. This service is most useful with the transfer and conference services.

The call hold service must be enabled for this service to work. See [“Call Hold” on page 102](#).

► To enable the Second Call service:

1. Select to which port you want to apply the changes in the drop down menu at the top of the window.
2. In the *Second Call* section, enable the service by selecting **Enable** in the *Activation* choice.

Figure 57: Second Call Section

Second Call	
Activation:	☒ Enable ☐ Disable
Status:	Active

3. Click *Submit* if you do not need to set other parameters.

Using Second Call

The following is the procedure to use this service on the user's telephone.

► To use the second call service:

1. Perform a Flash-Hook by pressing the “Flash” button on your analog telephone.
This puts the call on hold and the second line is automatically connected to your line.
2. Initiate the second call.

Automatic Call

The automatic call feature allows you to define a telephone number that is automatically dialed when taking the handset off hook.

When this service is enabled, the second line service is disabled but the call waiting feature is still functional. The user can still accept incoming calls.

► To set the automatic call feature:

1. In the web interface, click the *Telephony* link, then the *Services* sub-link.
This links to the *Telephony – Advanced* web page.

Figure 58: Telephony – Advanced Web Page

2. Select to which port you want to apply the changes in the drop down menu at the top of the window.
3. In the *Automatic Call* section, enable the service by selecting **Enable** in the *Automatic Call Activation* choice.
4. Define the number to dial when the handset is taken off hook in the *Automatic Call Target* field.
Accepted formats are:
 - telephone numbers (5551111)
 - SIP URLs such as "scheme:user@host". For instance, "sip:user@foo.com".
 This string is used literally, so cosmetic symbols (such as the dash in "555-xxxx") should not be present.
5. Click *Submit* if you do not need to set other parameters.

Miscellaneous

The *Misc* sub-page of the *Telephony* page allows you to set the following parameters:

- ▶ Country
- ▶ Custom Tone Configuration
- ▶ Message Waiting Indicator

Country Selection

It is very important to set the country in which the Mediatrix 4100 is used because a number of parameter values are set according to this choice. These parameters are:

- ▶ Tones
- ▶ Rings
- ▶ Impedances
- ▶ Line Attenuations

See ["Appendix D - Country-Specific Parameters" on page 385](#) for more information on these country-specific settings. You can also set these parameters via SNMP, as described in ["Setting the Location \(Country\)" on page 178](#).

▶ To set a country location:

1. In the web interface, click the *Telephony* link, then the *Misc* sub-link.

Figure 59: Telephony – Misc Web Page

2. In the *Country* section, select the country in which the Mediatrix 4100 is located in the *Country Selection* field.

3. Click *Submit* if you do not need to set other parameters.
This parameter is set for all the lines of the Mediatrix 4100. You cannot have a different country for each line.

Caller ID Selection

In countries that support more than one caller ID standard, this standard can be selected with the *Country Selection* field. Be careful to properly select the option corresponding to your caller ID.

Table 61: Caller ID Mappings

Country	Caller ID	<i>Country Selection</i> field Mapping
UK	British Telecom	uk
	Bellcore	uk-bellcore
	CCA	uk-cca
	ETSI-FSK	uk-etsi-fsk
France	Bellcore	france
	ETSI-FSK	france-etsi-fsk
	ETSI-DTMF	france-etsi-dtmf
Austria1	Bellcore	austria1
	ETSI-FSK	austria-etsi-fsk
Austria2	Bellcore	austria2
	ETSI-FSK	austria2-etsi-fsk

See ["Caller ID Information" on page 177](#) for more details.

Custom Tone Configuration

You can override the pattern for a specific tone defined for the selected country (see ["Appendix D - Country-Specific Parameters" on page 385](#) for more details). You can define new patterns for the following tones:

- | | |
|-------------------|----------------------------------|
| ▶ Busy | ▶ Preemption |
| ▶ Confirmation | ▶ Reorder |
| ▶ Congestion | ▶ Ringback |
| ▶ Dial | ▶ Receiver Off Hook (ROH) |
| ▶ Intercept | ▶ Special Information Tone (SIT) |
| ▶ Message Waiting | ▶ Stutter |

Pattern Definition

The general format of the pattern string is defined in the following ABNF:

```
tone-pattern = [ frequencies-section [ "," loop-counter-section ] "," states-section ]
```

This general pattern uses the following three main categories

```
frequencies-section = "f1=" frequency-description
                    [ ",f2=" frequency-description
                    [ ",f3=" frequency-description
                    [ ",f4=" frequency-description ] ] ]

loop-counter-section = "l=" loop-counter

states-section = ",s1=" state-description
               [ ",s2=" state-description
               [ ",s3=" state-description
               [ ",s4=" state-description
               [ ",s5=" state-description
               [ ",s6=" state-description
               [ ",s7=" state-description
               [ ",s8=" state-description ] ] ] ] ] ] ]
```

Finally, the three main categories use the following parameters and tags:

```
frequency-description = frequency ":" power
frequency = 2*4DIGIT
power = ( DIGIT / "-" 1*2DIGIT )
loop-counter = 1*2DIGIT
state-description = ( on-state-description / off-state-description )
on-state-description = "on" frequency-selection [ time ] [ loop-indicator ] [ next-state ]
off-state-description = "off" [ time ] [ loop-indicator ] [ next-state ]
frequency-selection = [ ":f1" ] [ ":f2" ] [ ":f3" ] [ ":f4" ]
time = ":" 2*5DIGIT
loop-indicator = "l"
next-state = ":" ( "s1" / "s2" / "s3" / "s4" / "s5" / "s6" / "s7" )
```

The following table describes the various tags used in the syntax.

Table 62: Pattern Definition Syntax

Tag	Description
tone-pattern	String describing the pattern to use for the tone. An empty string means no tone.
frequencies-section	Description of the frequencies used by the tones used in states-section. You can define up to four frequencies (f1 to f4). You must enter at least one frequency if the tone-pattern is not empty. The frequencies to use are defined in the state-description.
frequency-description	Description of the frequency. This is described as "frequency:power".
frequency	Frequency value in Hz. The range is from 10 Hz to 4000 Hz.
power	Power level of the frequency in dBm. The range is from -99 dBm to 3 dBm.
loop-counter-section	Loop counters definition. The loop counter is used in state-description.
loop-counter	Value of the loop counter. The range is from 2 to 128.
states-section	Description of the tone state. You can define up to eight states (s1 to s8). You must enter at least one state if the tone-pattern is not empty.
state-description	Description of the tone state.

Table 62: Pattern Definition Syntax (Continued)

Tag	Description
on-state-description	Description of a state playing a tone.
off-state-description	Description of a state not playing a tone.
frequency-selection	Frequency to play in the state. You can use from one to four frequencies. The frequency must be defined in frequencies-section.
time	The number of times, in ms, to perform the action of the state. The range is from 10 ms to 56000 ms. The tone stays indefinitely in the state if no time is specified.
loop-indicator	Used to stop looping between states after a number of loops defined in loop-counter-section. When the number of loops is reached, the next state is s(n+1) for the state s(n) instead of the state defined in next-state.
next-state	The next tone state to use when the time has elapsed. This value is not present if the time is not present.

Customizing the Tones

The *Custom Tone* section allows you to define new patterns as per the pattern syntax.

► To customize one or more tones:

1. In the *Custom Tone* section, of the *Misc* page, define whether or not you want to override the default tone configuration for a specific tone by setting the corresponding *Override* column.

Figure 60: Custom Tone Section

Custom Tone	Override	Pattern
Busy	Disable	f1=350-17,f2=440-17,s1=on,f1,f2
Confirmation	Disable	
Congestion	Disable	
Dial	Disable	f1=200-15,f2=300-20,f3=400-25,f4=500-30,s1=on,f1,f2,s2=on,f1,f2
Intercept	Disable	
Message Waiting	Disable	
Preemption	Disable	
Reorder	Disable	
Ringback	Disable	
ROH	Disable	
SIT	Disable	
Stutter	Disable	



Note: The default hold tone value is mute (i.e., no tone).

2. Enter the override pattern in the corresponding *Pattern* column.
You must follow the syntax described in [“Pattern Definition” on page 106](#).
See [“Custom Tone Example” on page 108](#) for a detailed example on how to create a proper pattern.
The following table gives some examples of custom tones. Note that the quotation marks are not part of the syntax and must not be included when entering the tone pattern.

Table 63: Pattern Examples

Example	Pattern
No tone	""

Table 63: Pattern Examples (Continued)

Example	Pattern
North America dial tone (continuous tone at 350 Hz and 440 Hz with a -17 dBm power level)	"f1=350:-17,f2=440:-17,s1=on:f1:f2"
North America Recall dial tone (three quick tones followed by a continuous tone)	"f1=350:-17,f2=440:-17,l=3,s1=on:f1:f2:100:s2,s2=off:100:l:s1,s3=on:f1:f2"
Australia ring back tone (tone on 400 ms, off 200 ms, on 400 ms, and off 2000 ms and replay)	"f1=425:-17,f2=400:-5,f3=450:-5,s1=on:f1:f2:f3:400:s2,s2=off:200:s3,s3=on:f1:f2:f3:400:s4,s4=off:2000:s1"

- Click *Submit* if you do not need to set other parameters.

Custom Tone Example

This section describes how to create the pattern for the North America recall dial tone (also called stutter dial tone), which is three quick tones followed by a continuous tone.

```
"f1=350:-17,f2=440:-17,l=3,s1=on:f1:f2:100:s2,s2=off:100:l:s1,s3=on:f1:f2"
```

► To create the pattern:

- Let's start with the general format of the pattern string:

```
Pattern = [ frequencies-section [ "," loop-counter-section ] "," states-section ]
```

- Set the frequencies-section category, which is defined as follows:

```
frequencies-section = "f1=" frequency-description
                    [ ",f2=" frequency-description
                    [ ",f3=" frequency-description
                    [ ",f4=" frequency-description ] ] ]
```

- The frequency-description parameter is described as follows:

```
frequency:power
```

- In the North America stutter dial tone, two frequencies are used, 350 Hz and 440 Hz. Their power level is -17 dBm. You can thus complete the frequencies-section category as follows:

```
frequencies-section = "f1=" 350:-17 ",f2=" 440:-17 ]
```

- The general format of the pattern string now looks as follows:

```
Pattern = [ f1=350:-17,f2=440:-17 [ "," loop-counter-section ] "," states-section ]
```

- Set the loop-counter-section category, which is defined as:

```
loop-counter-section = "l=" loop-counter
```

It defines the number of times to repeat the pattern.

- a. The loop-counter part is defined as follows:

```
loop-counter = 1*2DIGIT
```

- b. In the North America stutter dial tone, the pattern is repeated three times, thus:

```
loop-counter = 3
```

- c. The loop-counter-section category now looks as follows:

```
loop-counter-section = "l=" 3
```

- d. The general format of the pattern string now looks as follows:

```
Pattern = [ f1=350:-17,f2=440:-17 [ "," "l=" 3 ] "," states-section ]
```

4. Set the states-section category, which is defined as:

```
states-section = ",s1=" state-description
                [ ",s2=" state-description
                [ ",s3=" state-description
                [ ",s4=" state-description
                [ ",s5=" state-description
                [ ",s6=" state-description
                [ ",s7=" state-description
                [ ",s8=" state-description ] ] ] ] ] ]
```

- a. state-description is defined as:

```
state-description = ( on-state-description / off-state-description )
```

- b. There are three states in the North America stutter dial tone: 0.1 on, 0.1 off, and continuous. The pattern that must be described is thus:

```
states-section = ",s1=" state-description
                [ ",s2=" state-description
                [ ",s3=" state-description ] ]
```

5. Let's define the first state. Since the first state describes an on tone, off-state-description is not required for this state.

- a. You now have to complete the on-state-description parameter for the first state, which is defined as:

```
on-state-description = "on" frequency-selection [ time] [ loop-indicator ] [
next-state ]
```

- b. frequency-selection is defined as the frequencies to play and has the following syntax:

```
frequency-selection = [ ":f1" ] [ ":f2" ] [":f3" ] [ ":f4" ]
```

You can use from one to four frequencies. The North America stutter dial tone has two frequencies, thus:

```
frequency-selection = [ ":f1" ] [ ":f2" ]
```

- c. The on-state-description parameter is now:

```
on-state-description = "on" [ ":f1" ] [ ":f2" ] ] [ time] [ loop-indicator ]
[ next-state ]
```

- d. The time parameter is defined as:

```
":" 2*5DIGIT
```

It is the number of milliseconds to perform the action of the state. The on state is 100 ms, thus,

```
on-state-description = "on" [ ":"f1" ] [ ":"f2" ] [ :100 ] [ loop-indicator ] [
next-state ]
```

- e. The loop-indicator parameter is not used in this state. The on-state-description parameter is now:

```
on-state-description = "on" [ ":"f1" ] [ ":"f2" ] [ :100 ] [ next-state ]
```

- f. The next-state parameter is defined as:

```
next-state = ":" ( "s1" / "s2" / "s3" / "s4" / "s5" / "s6" / "s7" )
```

It is the next tone state to use when the time has elapsed. In this case, the next-state parameter is the off state, which is designated as s2.

The on-state-description parameter is now:

```
on-state-description = "on" [ ":"f1" ] [ ":"f2" ] [ :100 ] [ :s2 ]
```

- g. You can now complete the first state-description parameter:

```
states-section = ",s1=" ( on-state-description)
                  [ ",s2=" state-description
                  [ ",s3=" state-description ] ]
```

becomes:

```
states-section = ",s1=" ( "on" [ ":"f1" ] [ ":"f2" ] [ :100 ] [ :s2 ] )
                  [ ",s2=" state-description
                  [ ",s3=" state-description ] ]
```

6. Let's define the second state. Since the first state describes an off tone, on-state-description is not required for this state.

- a. You now have to complete the off-state-description parameter for the first state.

The off-state-description parameter is defined as:

```
off-state-description = "off" [ time ] [ loop-indicator ] [ next-state]
```

- b. The time parameter is defined as:

```
":" 2*5DIGIT
```

It is the number of milliseconds to perform the action of the state. The off state is 100 ms, thus,

```
off-state-description = "off" [ :100 ] [ loop-indicator ] [ next-state]
```

- c. The loop-indicator parameter is defined as:

```
loop-indicator = ":l"
```

It is used to stop looping between states. It indicates that the loop stops after three times. Once the loop is completed, the pattern goes to the next state (which is state 3).

The off-state-description parameter is now:

```
off-state-description = "off" [ :100 ] [ :1 ] [ next-state ]
```

- d. The next-state parameter is defined as:

```
next-state = ":" ( "s1" / "s2" / "s3" / "s4" / "s5" / "s6" / "s7" )
```

It is the next tone state to use when the time has elapsed. In this case, the next-state parameter is the on state, which is designated as s1.

The off-state-description parameter is now:

```
off-state-description = "off" [ :100 ] [ : 1 ] [ :s1 ]
```

- e. You can now complete the second state-description parameter:

```
states-section = ",s1=" ( "on" [ ":f1" ] [ ":f2" ] [ :100 ] [ :s2 ] )
                  [ ",s2=" state-description
                    [ ",s3=" state-description ] ]
```

becomes:

```
states-section = ",s1=" ( "on" [ ":f1" ] [ ":f2" ] [ 100 ] [ s2 ] )
                  [ ",s2=" "off" [ :100 ] [ :1 ] [ :s1 ]
                    [ ",s3=" state-description ] ]
```

7. Let's define the third and last state. Since the third state describes an on tone, off-state-description is not required for this state.

- a. You now have to complete the on-state-description parameter for the first state.

The on-state-description parameter is defined as:

```
on-state-description = "on" frequency-selection [ time ] [ loop-indicator ] [
next-state ]
```

- b. frequency-selection is defined as the frequencies to play and has the following syntax:

```
frequency-selection = [ ":f1" ] [ ":f2" ] [ ":f3" ] [ ":f4" ]
```

You can use from one to four frequencies. The North America stutter dial tone has two frequencies, thus:

```
frequency-selection = [ ":f1" ] [ ":f2" ]
```

- c. The on-state-description parameter is now:

```
on-state-description = "on" [ ":f1" ] [ ":f2" ] [ time ] [ loop-indicator ] [
next-state ]
```

- d. The time parameter is the number of milliseconds to perform the action of the state. Since the third state is a continuous tone, the time parameter is not required, thus,

```
on-state-description = "on" [ ":f1" ] [ ":f2" ] [ loop-indicator ] [ next-
state ]
```

- e. The loop-indicator parameter is used to stop looping between states. Since the third state is a continuous tone and does not use loops, this parameter is not required.

The on-state-description parameter is now:

```
on-state-description = "on" [ ":f1" ] [ ":f2" ] [ next-state ]
```

- f. The next-state parameter is the next tone state to use when the time has elapsed. This value is not present if the time parameter is not present. You have already discarded the time parameter, so the next-state parameter is not required.

The on-state-description parameter is now:

```
on-state-description = "on" [ ":f1" ] [ ":f2" ]
```

- g. You can now complete the third state-description parameter and the states-section parameter:

```
states-section = ",s1=" ( "on" [ ":f1" ] [ ":f2" ] [ :100 ] [ :s2 ] )
                  [ ",s2=" "off" [ :100 ] [ :1 ] [ :s1 ]
                  [ ",s3=" state-description ] ]
```

becomes:

```
states-section = ",s1=" ( "on" [ ":f1" ] [ ":f2" ] [ :100 ] [ :s2 ] )
                  [ ",s2=" "off" [ :100 ] [ :1 ] [ :s1 ]
                  [ ",s3=" "on" [ ":f1" ] [ ":f2" ] ] ]
```

8. Now that you have the three main categories completed, you can finish the pattern:

```
Pattern = [ f1=350:-17,f2=440:-17 [ ", " "l=" 3 ] ", " ",s1=" ( "on" [ ":f1" ]
[ ":f2" ] [ :100 ] [ :s2 ] )
            [ ",s2=" "off" [ :100 ] [ :1 ] [ :s1 ]
            [ ",s3=" "on" [ ":f1" ] [ ":f2" ] ] ] ]
```

If you remove all the brackets and quotation marks, which are not to be included, the pattern is:

```
Pattern = f1=350:-17,f2=440:-
17,l=3,s1=on:f1:f2:100:s2,s2=off:100:1:s1,s3=on:f1:f2
```

Ther pattern could also be defined as follows:

```
Pattern = f1=350:-17,f2=440:-
17,s1=on:f1:f2:100:s2,s2=off:100:s3,s3=on:f1:f2:100:s4,s4=off:100:s5,s5=on:f
1:f2:100:s6,s6=off:100:s7,s7=on:f1:f2
```

Message Waiting Indicator

The Message Waiting Indicator (MWI) service alerts the user when new messages have been recorded on a voice mailbox.

When the user receives a call and does not answer, the notification mechanism detects this situation and starts the auto attendant. The caller can then leave a message.

After the message is recorded, the server sends a message to the Mediatrix 4100 listing how many new and old messages are available. The Mediatrix 4100 alerts the user of the new message in two different ways:

- ▶ The telephone's LED blinks (if present).
- ▶ A message waiting stutter dial tone replaces the normal dial tone when the user picks up the first line.



Note: The message waiting state does not affect the Second Line feature. When in an active call, performing a flash-hook to get access to the second line plays the usual dial tone.

Standard MWI Methods

The Mediatrix 4100 supports two MWI methods.

MWI Method #1

Standards Supported

- draft-ietf-sipping-mwi-01.txt (MWI draft)
- “Telecordia GR-1401-CORE (Issue 1, June 2000)” specification (visual message indication (LED blinking))
- “GR-506-CORE (Issue 1, with Revision 1, November 1996)” specification (message waiting indicator tone)

The Mediatrix 4100 sends SUBSCRIBE requests to the server for each line, unless there is no subscription address defined. The Mediatrix 4100 then waits for NOTIFY requests containing the relevant message waiting information.

You can also set these parameters via SNMP, as described in [“Chapter 25 - Message Waiting Indicator” on page 331](#).

► To configure the MWI service:

1. Select to which port you want to apply the changes in the drop down menu at the top of the window.
2. In the *Message Waiting Indicator* section, define the digits that users must dial to retrieve messages in the *Voice-Mail System Digit Map* field.

Dialing these digits initiates a call to the voice messaging system. For instance, you could decide to put “*50” as the sequence a user must dial to retrieve voice messages. This sequence must be unique and follow the syntax for digit maps (see [“Digit Maps” on page 73](#)). Dialing this digit map does not have any effect unless the service's status is “enabled”.

The activating sequence is set for all the lines of the Mediatrix 4100. You cannot have different sequences for each line.

Figure 61: Message Waiting Indicator Section

3. Set the destination to call to retrieve messages in the *Voice-Mail System Address* field.
The user typically initiates a call to the voice messaging system, and then uses an auto-attendant to get the messages. Available formats are:
 - telephone numbers (5551111)
 - SIP URLs such as “scheme:user@host”. For instance, “sip:user@foo.com”.

This string is used literally, so cosmetic symbols (such as the dash in “555-xxxx”) should not be present.

4. Set the notification mechanism server address to which the Mediatrix 4100 subscribes in the *MWI Subscription Address* field.

This mechanism notifies the Mediatrix 4100 when new messages are available. The address is a SIP URL such as “scheme:user@host”. For instance, “sip:user@foo.com”.

5. Set the subscription refresh in the *MWI Refresh* field.

Available values are:

- noOp: No operation.
- refresh: Refresh message waiting subscriptions. All enabled endpoints unsubscribe themselves from the service and re-subscribe by using the current provisioning.

This parameter is set for all the lines of the Mediatrix 4100. You cannot have a different behaviour for each line.

6. Define the duration, in seconds, of dynamic subscription to a messaging service in the *MWI Expiration Time* field.
This parameter is set for all the lines of the Mediatrix 4100. You cannot have a different behaviour for each line.
7. Enable the MWI by selecting **Active** in the *MWI Activation* field.
The MWI subscription refresh is not supported when the caller ID is DTMF-based.
8. Click *Submit* if you do not need to set other parameters.

MWI Method #2

Standards Supported	draft-mahy-sip-message-waiting-02.txt (expired) with proprietary modifications
----------------------------	--

This method does not require any special settings or configuration.

MWI Notify Service

The Mediatrix 4100 offers the possibility to extend some key features to remote extensions located in Branch or Home Offices across the SCN.

This service is available only when using the IP Communication Server v3.1 product as a SIP Redirect server. For instance, a designated analog voice mail system at a main site can provide voice mail for the home or branch office. The home office user is notified of the message waiting via a message waiting LED on the telephone or a special tone when picking up the telephone.

How does the Service Work?

The MWI Notify service is a proprietary feature. In this solution, the analog voice mail system is configured to seize a designated outgoing line and dial a pre-defined string such as “*72xxx” to notify the server it must give a message waiting indication to extension “xxx”. Once voice messages have been retrieved, the analog voice mail system seizes the designated outgoing line and dials a pre-defined string such as “*73xxx” to notify the server to turn off the message waiting indicator for extension “xxx”.

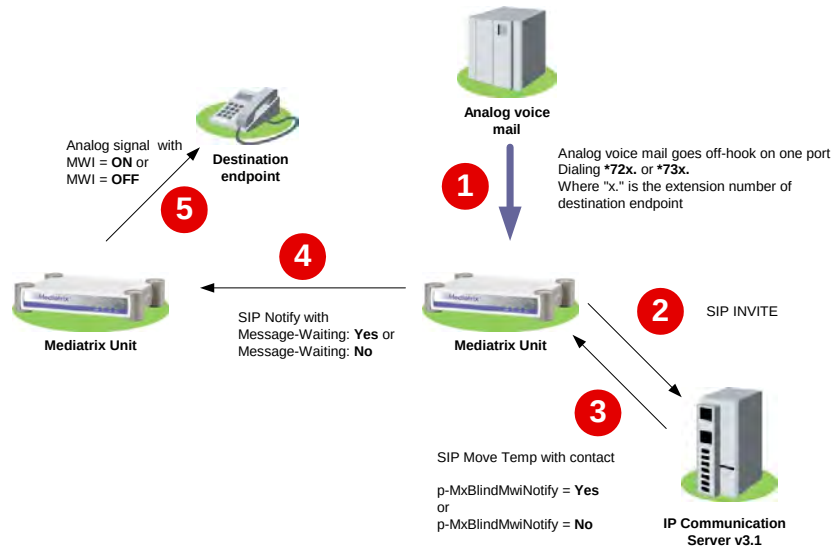
The service uses the Route Manager currently available in the IP Communication Server v3.1 to send a special command to the Mediatrix unit.

The following is the basic sequence of operations for the MWI Notify service:

1. The analog voice mail system dials the following digits:
*72101
where *72 is a prefix and 101 the user extension.
2. The Mediatrix unit sends a standard INVITE to the IP Communication Server v3.1 containing the complete dialed string (*72101).
 - a. The IP Communication Server looks for the registered user “*72101” in the Registrar database.
 - b. The IP Communication Server cannot find the user, so it asks the Route Manager to process the request.
 - c. Provided that the Route Manager is properly configured, it recognizes the “*72” prefix and associates it to the proper route conditions.
3. The IP Communication Server answers the request with a “Moved Temporarily”. It contains information about the target(s) in the *Contact* header plus a proprietary *p-MxBlindMWINotify=yes/no* field.
 - a. The Mediatrix unit retrieves the location from the IP Communication Server’s answer and the *p-MxBlindMWINotify* field.
 - b. The Mediatrix unit parses the answer from the IP Communication Server and recognizes *p-MxBlindMWINotify* as a special command.
4. The Mediatrix unit sends a NOTIFY to the location received from the IP Communication Server by using the proper yes or no value (*72 = yes, *73 = no) specified by the route condition.

5. The unit receiving the NOTIFY enables or disables the MWI service for the specified port/user.

Figure 62: Example of the MWI Notify Service



Configuring the IP Communication Server

In the Route Manager of the IP Communication Server, you must configure routes that would be triggered by a pre-defined prefix. The prefix could be any valid digits (DTMF). The example described above uses "*72" to enable the MWI and "*73" to disable the MWI.

For more information on how to configure the Route Manager, please refer to the *IP Communication Server Administration Manual* or the IP Communication Server contextual help.

Configuring the Mediatrix 4100

There is no special unit configuration required. The Mediatrix unit behaves as if in a standard call until it receives one of the following parameters in the *Contact* field:

- ▶ p-MxBlindMwiNotify=Yes
- or
- ▶ p-MxBlindMwiNotify=No

Upon receiving one of these parameters, the unit sends a NOTIFY to the destination endpoint instead of an INVITE. The sent NOTIFY is compliant with <draft-mahy-sip-message-waiting-02.txt>.

The *Advanced* page allows you to configure various system and network parameters of the Mediatrix 4100.

Quality of Service (QoS)

QoS (Quality of Service) features enable network managers to decide on packet priority queuing. The Mediatrix 4100 supports the Differentiated Services (DS) field and 802.1q taggings. The Mediatrix 4100 supports the Real Time Control Protocol (RTCP), which is used to send packets to convey feedback on quality of data delivery.

The Mediatrix 4100 does not support RSVP (Resource Reservation Protocol).

802.1q Configuration

The 802.1q standard recommends the use of the 802.1q VLAN tags for Ethernet frames traffic prioritization. VLAN tags are 4-byte headers in which three bits are reserved for priority indication. The values of the priority bits shall be provisioned.

The 802.1q standard comprises the 802.1p standard.

You can also set these parameters via SNMP, as described in [“Chapter 27 - Quality of Service \(QoS\)” on page 337](#).

VLANs

VLANs are created with standard Layer 2 Ethernet. A VLAN Identifier (VID) is associated with each VLAN. VLANs offer the following benefits:

- VLANs are supported over all IEEE 802 LAN MAC protocols, and over shared media LANs as well as point-to-point LANs.
- VLANs facilitate easy administration of logical groups of stations that can communicate as if they were on the same LAN. They also facilitate easier administration of moves, adds, and changes in members of these groups.
- Traffic between VLANs is restricted. Bridges forward unicast, multicast, and broadcast traffic only on LAN segments that serve the VLAN to which the traffic belongs.

The VLAN field in the Ethernet frame is located after both destination and source addresses:

```

0                               1
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7   (byte)
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| Dest Addr | Src Addr | VLAN | Type/Length | ...
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

The VLAN field is separated as follows:

```

0 (bit)           1           2           3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                               0x8100                               | Pri | T |                               VID                               |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

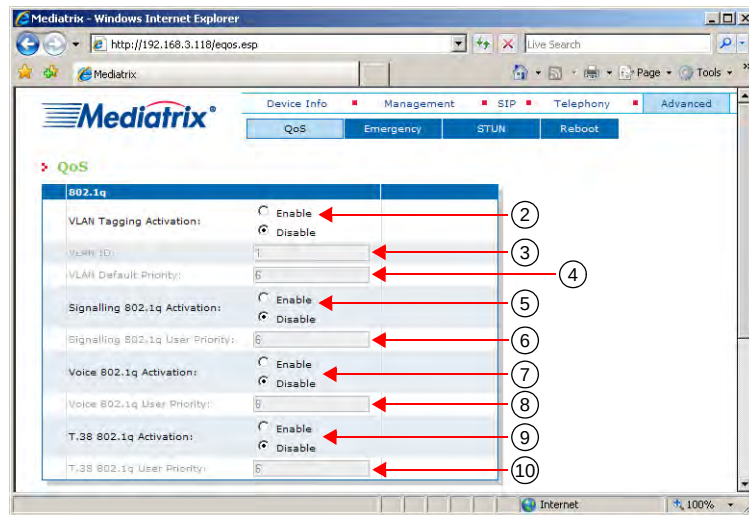
```

For both signalling and media packets, the VLAN priority section is configurable independently.

► To enable the IEEE 802.1q user priority configuration:

1. In the web interface, click the *Advanced* link, then the *QoS* sub-link.

Figure 63: Advanced – QoS Web Page



2. Enable the VLAN tagging by selecting **Enable** in the *VLAN Tagging Activation* field.
All packets are tagged with the Virtual ID (VID) specified in the *VLAN ID* field.
Enable this option only on compatible LAN with equipment supporting the VLAN tagging. Otherwise, the Mediatrix 4100 may be unreachable. In this case, use the *Reset / Default* button to access and disable VLAN tagging – in the recovery mode, tagging is not permitted.
3. Set the 802.1Q Virtual LAN ID in the *VLAN ID* field.
This is the VID to be applied in the TCI field when tagging is enabled. The value 1 is the default Port VID (PVID) for bridge port. The 4095 VID (0xFFFF) is reserved and it must not be used in tag header.
4. Set the 802.1Q Virtual LAN default user priority in the *VLAN Default Priority* field.
This is the user priority to be applied in the TCI field when tagging is enabled. This value applies to all protocols for which no priority filtering is enabled (e.g. ARP, ICMP).
 - 7 = High priority
 - 0 = Low priority
5. Enable the 802.1Q VLAN user priority tagging for VoIP signalling packet by selecting **Enable** in the *Signalling 802.1q Activation* field.
This filter applies to any VoIP signalling protocol in use. Each signalling packet is tagged with the user priority defined in the *Signalling 802.1q User Priority* field.
6. Set the IEEE 802.1Q VLAN user priority value for VoIP signalling packet in the *Signalling 802.1q User Priority* field.
 - 7 = High priority
 - 0 = Low priority
7. Enable 802.1Q VLAN user priority tagging for VoIP packet by selecting **Enable** in the *Voice 802.1q Activation* field.
This filter applies to any VoIP voice protocol in use (e.g. RTP). Each signalling packet is tagged with the user priority defined in the *Voice 802.1q User Priority* field.
8. Set the 802.1Q VLAN user priority value for VoIP packet in the *Voice 802.1q User Priority* field.
 - 7 = High priority
 - 0 = Low priority

9. Enable 802.1Q VLAN user priority tagging for T.38 fax packet by selecting **Enable** in the *T.38 802.1q Activation* field.
Each signalling packet is tagged with the user priority defined in the *T.38 802.1q User Priority* field.
10. Set the 802.1Q VLAN user priority value for T38 fax packet in the *T.38 802.1q User Priority* field.
 - 7 = High priority
 - 0 = Low priority
11. Click *Submit* if you do not need to set other parameters.

VLAN User Priority

The VLAN user priority values are used to set the user priority in the TCI field of the VLAN tag. Tagging user priority is applied only when the filter is enabled. When the filter for signalling protocol is disabled and the VLAN option is enabled, the Mediatrix 4100 uses the default user priority defined in the *VLAN Default Priority* field. Otherwise, the user priority set for signalling has precedence over the VLAN default user priority.

DiffServ Configuration

Standards Supported

RFC 2475 – An Architecture for Differentiated Services

Differentiated Services (DiffServ, or DS) is a protocol for specifying and controlling network traffic by class so that certain types of traffic – for example, voice traffic, which requires a relatively uninterrupted flow of data, might get precedence over other kinds of traffic.

DiffServ replaces the first bits in the ToS byte with a differentiated services code point (DSCP). It uses the existing IPv4 Type of Service octet.

► To set the DiffServ configuration:

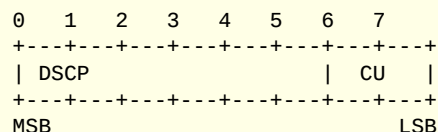
1. In the *DiffServ* section of the *QoS* page, set the following DiffServ fields:
 - Voice DiffServ Value:
 - T.38 DiffServ Value
 - Signalling DiffServ Value

These fields are 1 octet scalar ranging from 0 to 255. The DSCP default value should be 101110. This results in the DS field value of 10111000 (184d).

What are Differentiated Services?

Differentiated Services avoids simple priority tagging and depends on more complex policy or rule statements to determine how to forward a given network packet. An analogy is made to travel services, in which a person can choose among different modes of travel – train, bus, airplane – degree of comfort, the number of stops on the route, standby status, the time of day or period of year for the trip, and so forth.

For a given set of packet travel rules, a packet is given one of 64 possible forwarding behaviors – known as per hop behaviors (PHBs). A six-bit field, known as the Differentiated Services Code Point (DSCP), in the Internet Protocol header specifies the per hop behavior for a given flow of packets. The DS field structure is presented below:



- *DSCP*: Differentiated Services CodePoint.
- *CU*: Currently Unused. The CU bits should always be set to 0.

For both signalling and media packets, the DSCP field is configurable independently. The entire DS field (TOS byte) is currently configurable.

This default value would result in a value of “101” precedence bits, low delay, high throughput, and normal reliability in the legacy IP networks (RFC 791, RFC 1812). Network managers of legacy IP networks could use the above-mentioned values to define filters on their routers to take advantage of priority queuing. The default value is based on the Expedited Forwarding PHB (RFC 2598) recommendation.



Note: RFC 3168 now defines the state in which to set the two least significant bits in the TOS byte. On the other hand, this RFC only applies to TCP transmissions and the bits are thus set to “0” in the Mediatrix 4100. This has the following effects:

- The TOS values for UDP packets are the same as in the MIB.
- The TOS values for TCP packets are equal to the closest multiple of 4 value that is not greater than the value in the MIB.

Figure 64: DiffServ Section

2. Click *Submit* if you do not need to set other parameters.

Emergency Page

The *Emergency* sub-page of the *Misc* page allows you to configure the Emergency Call parameters of the Mediatrix 4100.

Emergency Call Configuration

The Emergency Call service (also called urgent gateway) allows a “911”-style service. It allows a user to dial a special digit map resulting in a message being sent to a specified urgent gateway, bypassing any other intermediaries.

If enabled, whenever the user dials the specified digit map, a message is sent to the target address.

You can also set these parameters via SNMP, as described in [“Emergency Call” on page 304](#).

► To configure the emergency call service:

1. In the web interface, click the *Advanced* link, then the *Emergency* sub-link.

Figure 65: Advanced – Emergency Web Page

2. Enable the emergency call feature by selecting **Enable** in the *Emergency Call Activation* choices.
3. Set the number to reach for an urgent call in the *Emergency Call Target* field.
Accepted formats are:
 - telephone numbers (5551111)

- SIP URLs such as "scheme:user@host". For instance, "sip:user@foo.com".

Note that this string is used literally, so cosmetic symbols (such as the dash in "555-xxxx") should not be present.

4. Define the digits that users must dial to start the urgent gateway call feature in the *Emergency Call Digit Map* field.

For instance, you could decide to put "*60" as the sequence a user must dial to start the urgent gateway service. This sequence must follow the syntax for digit maps (see ["Digit Maps" on page 73](#)). Dialing this digit map does not have any effect unless the service's status is "enabled". The activating sequence is set for all the lines of the Mediatrix 4100. You cannot have different sequences for each line.

5. Click *Submit* if you do not need to set other parameters.

STUN Configuration

Standards Supported

RFC 3489 – STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs)

STUN (Simple Traversal of UDP through NATs) is a simple client / server protocol that uses UDP packets to discover the configuration information of NATs and firewalls between a device and the public Internet:

- ▶ NAT type
- ▶ NAT binding public address
- ▶ NAT binding time to live

NAT (Network Address Translator) is a device that translates the IP address used within a "private" network to a different IP address known in another "public" network. See ["NAT Traversal" on page 264](#) for more details.

STUN supports a variety of existing NAT devices and does not require any additional hardware or software upgrades on the NAT device.

The Mediatrix 4100 uses the STUN protocol to discover its NAT binding for the following three IP addresses/ports (sockets):

- ▶ Signalling protocol (SIP) IP address/port
- ▶ RTP IP address/port
- ▶ T.38 IP address/port

SIP Outbound Proxy

For a unit to work properly behind a firewall, it must keep a pinhole opened by sending keepalive packets through the firewall.

The Mediatrix 4100 only sends keepalive packets to the last destination for a specific socket. When a unit is not configured with an outbound proxy, it can send, through its SIP socket, messages to various destinations, such as a SIP redirect server, another SIP unit, or a MWI server. If, for instance, the last SIP message was sent to the MWI server, the Mediatrix 4100 will keep the pinhole opened for the MWI server only (sending keepalive message to the MWI server) and won't be reachable by other units outside the firewall.

To avoid those issues, all SIP message should come and go from the same source/destination on the public side of the firewall, i.e., a SIP outbound proxy. Media5 thus recommends that you use a SIP outbound proxy. See ["SIP Servers Configuration" on page 63](#) for more details.

Restrictions on the Media5 STUN Implementation

- ▶ The Mediatrix 4100 does not currently support NAT type discovery.
- ▶ The Mediatrix 4100 does not currently support STUN NAT binding time to live discovery.
- ▶ The Mediatrix 4100 does not currently support the TLS security mechanism.

- Due to a limitation of most routers, an RTP portal might be required in order for two units behind the same NAT/firewall to be able to communicate with each other.

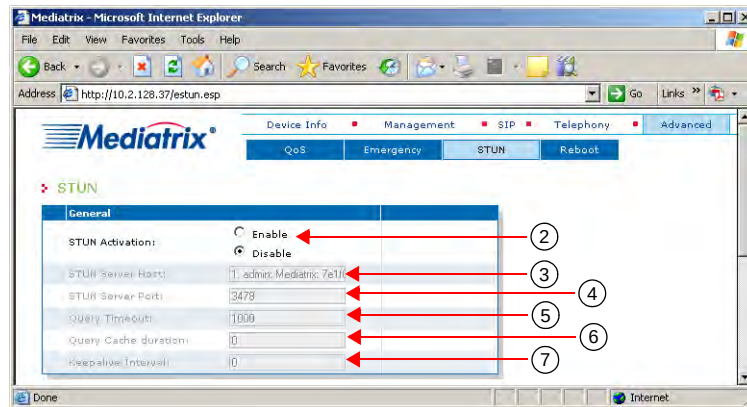
STUN Client Configuration

The *STUN* sub-page of the *Advanced* page allows you to configure the STUN client of the Mediatrix 4100. You can also set these parameters via SNMP, as described in [“Chapter 19 - STUN Configuration” on page 287](#).

► To set STUN parameters:

- In the *Advanced* pages, click the *STUN* link.

Figure 66: STUN Web Page



- Enable the STUN client by selecting the **Enable** option in the *STUN Activation* choices.
- Set the static STUN server IP address or Fully Qualified Domain Name (FQDN) in the *STUN Server Host* field.
- Set the static STUN server IP port number in the *STUN Server Port* field.
The default value is **3478**.
- Set the maximum amount of time, in milliseconds, the Mediatrix 4100 should wait for an answer to a STUN query sent to a STUN server in the *Query Timeout* field.
Available values range from 500 ms to 10000 ms.
Caution is advised in setting long timeouts. In the advent of an unresponsive STUN server, the unit may end up waiting a long time before it determines that a call cannot be made due to the STUN server failure.
- Set the amount of time, in seconds, the Mediatrix 4100 should keep a STUN query result in its internal cache in the *Query Cache duration* field.
Keeping a query in a cache helps to reduce the amount of unnecessary STUN queries when an RTP or T.38 socket is re-used within a short period of time. Available values range from 0 s to 3600 s.
When set to **0**, the cache is disabled and the unit performs a STUN query each time a socket needs to be used.
- Define the interval, in seconds, at which the Mediatrix 4100 sends blank keepalive messages to keep a firewall hole opened in the *Keepalive Interval* field.
Keepalive messages are used by both the signalling protocol socket and the RTP socket to keep those connections opened through a firewall. Available values range from 0 s to 120 s.
When set to **0**, no keepalive packet is sent.



Note: Keepalive messages are not supported on the T.38 socket.

- Click *Submit* if you do not need to set other parameters.

SIP Custom NAT Traversal

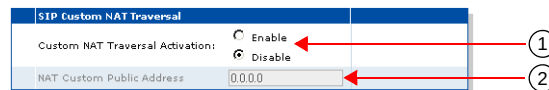
The Mediatrix 4100 may be used in a private domain that is not directly connected to the IP network. For instance, this may be the case for ITSP (Internet Telephony Service Provider) clients that have a small private network. This private network is connected to the public IP network through the NAT (Name Address Translation) technology.

You can configure the Mediatrix 4100 with the public IP address of the NAT system, which allows to reach the unit. SIP packets sent by the Mediatrix 4100 contain the NAT address configured as SIP contact. If the NAT service is not activated, the real IP address of the Mediatrix 4100 is used.

► **To set SIP custom NAT traversal parameters:**

1. In the *SIP Custom NAT Traversal* section of the *STUN* page, enable the custom NAT traversal by selecting the **Enable** option in the *Custom NAT Traversal Activation* choices.

Figure 67: SIP Custom NAT Traversal Section



SIP Custom NAT Traversal	
Custom NAT Traversal Activation:	☒ Enable ☐ Disable
NAT Custom Public Address	0.0.0.0

2. Enter the public IP address of the NAT system in the *NAT Custom Public Address* variable. This is the public IP address used as Contact address by outgoing SIP packets crossing a NAT system.
3. Click *Submit* if you do not need to set other parameters.

SNMP Configuration

Page Left Intentionally Blank

MIB Structure and SNMP

This chapter describes how the Mediatrix 4100 uses the SNMP protocol for its configuration.

SNMP Overview

The Mediatrix 4100 uses the Simple Network Management Protocol (SNMP) for initial software configuration provisioning and subsequent software configuration.

SNMP is a simple request-reply protocol for Internet network management services. It consists of *network management stations* (in this document, they are referred to as a management server) communicating with *network elements*. Management stations are normally workstations that display relevant facts about the elements being monitored.

SNMP works over the IP (Internet Protocol) communication stack. SNMP network management consists of three pieces:

- ▶ The protocol between the manager and the element (SNMP). This details the format of the packets exchanged. Although a wide variety of transport protocols could be used, UDP is normally used with SNMP.
- ▶ A set of common structures and an identification scheme used to reference the variables in the MIB. This is called the *Structure of Management Information (SMI)*.
- ▶ A *Management Information Base (MIB)* that specifies what variables the network elements maintain (the information that can be queried and set by the manager).

Definitions

Structure of Management Information (SMI)

The SMI is the set of rules for specifying the management information that a device maintains. The management information is actually a collection of managed objects, and these rules are used to both name and define these managed objects.

Management Information Base (MIB)

A MIB is a structured collection of all the managed objects a device maintains. The managed objects are structured in the form of a hierarchical tree. At the top of the tree is the most general information available about a network. Each branch of the tree then gets more detailed into a specific network area, with the leaves of the tree as specific as the MIB can get.

Object Identifier (OID)

Object Identifiers (OID) are strings of numbers. They are allocated in a hierarchical manner, so that, for instance, the authority for “1.2.3” is the only one that can say what “1.2.3.4” means. The formal definition of OIDs comes from ITU-T recommendation X.208 (ASN.1), which is available from the ITU.

SNMP Versions

The Mediatrix 4100 supports three versions of SNMP: SNMPv1, SNMPv2c and SNMPv3. SNMP defines a few types of messages that are exchanged between the manager and agent.

SNMPv1 Messages

The following messages are specific to SNMPv1.

Table 64: SNMPv1 Message Types

Operator	Description
messages sent from the manager to the agent	
get-request	Get the value of one or more variables.
get-next-request	Get the next variable after one or more specified variables.
set-request	Set the value of one or more variables.
messages sent from the agent to the manager	
get-response	Return the value of one or more variables. This is the message returned by the agent to the manager in response to the get-request , get-next-request , and set-request operators.
trap	Notify the manager when something happens on the agent.

SNMPv2c Messages

There are a few flavours of SNMPv2, SNMPv2c being the most common. The following message is specific to SNMPv2.

Table 65: SNMPv2 Message Type

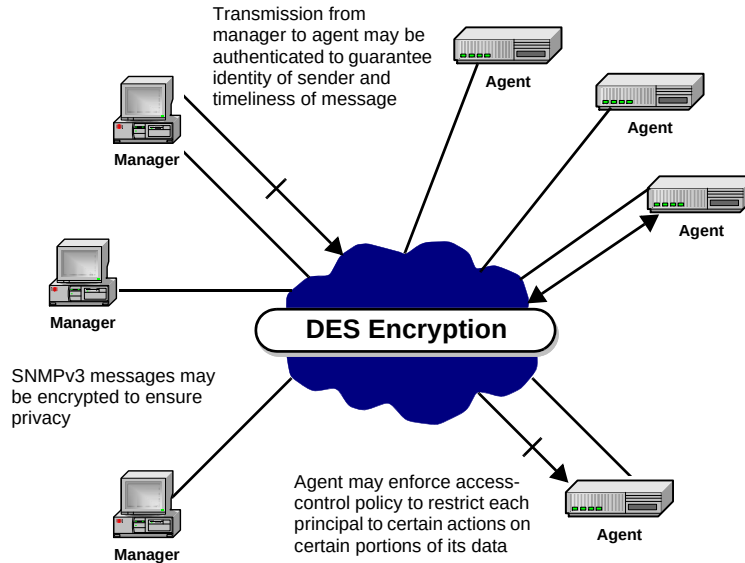
Operator	Description
get-bulk	Uses BULK Requests to query for a tree of information about a network entity. A variable put in command line specifies which portion of the object identifier space will be searched using BULK Requests. All variables in the subtree below the given variable are queried as a single request and their values presented to the user.

This message is sent from the manager to the agent.

SNMPv3 Messages

To correct the security deficiencies of SNMPv1/v2, SNMPv3 was defined with an overall SNMP architecture and a set of security capabilities. SNMPv3 includes three important services: *authentication*, *privacy*, and *access control* (Figure 68). To deliver these services in a flexible and efficient manner, SNMPv3 introduces the concept of a *principal*, which is the entity on whose behalf services are provided or processing takes place.

Figure 68: SNMPv3 Services



SNMP Behaviour

When using SNMP, the following rules apply:

- ▶ Media5 recommends to copy the SNMPv3 user attributes only twice.
- ▶ The administrator may edit the SNMPv3 user attributes:
 - Authentication algorithm (none, MD5, or SHA)
 - Authentication password
 - Encryption algorithm (NULL or DES)
 - Encryption password
 - All SNMPv3 passwords (encryption and authentication) must be at least 8 characters long. You should use the *Unit Manager Network* product to perform SNMPv3 setup. Whatever the MIB browser you use, the unit follows the SNMPv3 standard RFCs.

SNMP can be used in a non-secure or secure management mode.



Warning: The SNMPv3 method for changing the password or encryption key contains a flaw which may result in setting the incorrect password. This problem can happen if you use an incorrect “oldpassword” when changing your password. Always exercise great caution when changing your password or encryption key. Note that you can use the factory reset to clear the SNMPv3 password. See [“Factory Reset” on page 21](#) for more details. See also the *Unit Manager Network Administration Manual*.

Non-Secure Management Mode

In non-secure management mode, the unit responds to SNMP requests as follows:

- ▶ SNMPv1: read-write on all MIB tree
- ▶ SNMPv2c: read-write on all MIB tree
- ▶ SNMPv3: read-write on all MIB tree by using:
 - MD5 authentication
 - Authentication password: "Md5Password" (initial password)
 - DES encryption
 - Encryption password: "DesPassword" (initial password)
- ▶ SNMPv3: read-write on all MIB tree by using:
 - SHA authentication
 - Authentication password: "ShaPassword" (initial password)
 - DES encryption
 - Encryption password: "DesPassword" (initial password)

Secure Management Mode

In secure management mode, the unit responds to SNMP requests as follows:

- ▶ SNMPv1: read-only on all MIB tree
- ▶ SNMPv2c: read-only on all MIB tree
- ▶ SNMPv3: the same values as for SNMPv3 in non-secure management mode



Note: If you forget or lose a password, perform a Factory Reset to reset the unit to the non-secure management mode. See ["Factory Reset" on page 21](#) for more details.

Notes

- ▶ When using SNMPv3 with encryption (DES), you may experience delays when accessing MIB variables. This is normal because encrypting an IP packet takes in general longer than sending it over IP. If you experience any timeout, add some seconds to the timeout period of your MIB browser, and then try to reach the unit again.
- ▶ Suppose that the Mediatrix 4100 accepts requests with authentication only. If you perform requests by using encryption and authentication, assuming that the authentication password is valid, the SNMP agent still responds as if the requests were only authenticated.
- ▶ If you clone an SNMPv3 user, and then remove authentication or privacy for it, ensure that a row in *vacmGroupName* matches its new constraints. If not, the unit is not accessible by using the new clone parameters.

SNMPv3 Special Behaviour

Mediatrix units coming out of factory are set so that you can use all MIB variables by using SNMPv1, SNMPv2c, or SNMPv3. However, you can decide to accept only SNMPv3 access by using passwords known by administrators only for enhanced security. In this case, you should manually disable SNMPv1 / SNMPv2 so that SNMPv3 works properly. The Mediatrix 4100 thus refuses any SNMPv1 or SNMPv2 request it receives. You can disable / enable SNMPv1 / SNMPv2 by using the MIB Browser included in the Media5 Unit Manager Network (or any other MIB Browser) to modify the permissions related to SNMPv1 / SNMPv2 (security model). These permissions are located in the *VacmAccessTable* of the SNMP-VIEW-BASED-ACM-MIB (RFC 2575).

When using exclusively SNMPv3, a row from one of the following tables:

- ▶ *usmUserTable*
- ▶ *vacmSecurityToGroupTable*
- ▶ *vacmAccessTable*
- ▶ *vacmViewTreeFamilyTable*

is saved in flash memory only if these conditions are met:

- ▶ The RowStatus variable (e.g., *vacmAccessRowStatus*) is equal to **active(1)**.
- ▶ The StorageType variable (e.g., *vacmAccessStorageType*) is equal to **nonVolatile(3)**.



Note: The *vacmContextTable* is not saved under any condition.

SNMP Configuration via a Configuration File

You can modify the SNMP configuration of the Mediatrix 4100 by inserting an SNMP Agent section in a configuration file and then transferring this configuration file into the unit. This configuration replaces any configuration set in a profile. For more information on how to use a configuration file for updating the Mediatrix 4100, see [“Chapter 12 - Configuration File Download” on page 187](#).



Caution: The SNMP Agent section contains the default Media5 parameters related to SNMP. Default values enable SNMPv1, SNMPv2, and SNMPv3 and provide default Media5 credentials for SNMPv3.

The SNMP Agent section is located in the *SnmGenericTemplate.xml* file located under *Unit Manager Network 3.2\UnitManager\DefaultCfgFile* folder. The contents of the *SnmGenericTemplate.xml* file may be appended at the end of the generated XML file. See the Unit Manager Network documentation for more details.

The SNMP agent section must not be separated by other comments or OIDs in the configuration file.

If you transfer a configuration file with an SNMP Agent section that constitutes a change from the SNMPv3 configuration currently in use, the new configuration is applied and the unit then restarts so that the changes take effect.

A few notes:

- ▶ Once an SNMPv3 configuration is in effect in the Mediatrix 4100, it is not possible to revert the unit back to SNMPv1 or SNMPv2c by sending it a configuration file that does not include an SNMP Agent configuration section.
- ▶ If you perform a factory reset, all settings previously applied via the configuration file (including the SNMPv3 configuration) are lost and the unit reinitializes by using the current profile.

Figure 69: SNMP Agent Section Example

```
<SnmAgentConfig>
<snmpV2>
  <snmpModules>
    <snmpUsmMIB>
      <usmMIBObjects>
        <usmUser>
          <usmUserTable>
            <a>
              <usmUserSecurityName ValueType="OCTET_STRING" Value="Md5DesUser"/>
              <usmUserAuthProtocol ValueType="OID" Value="1.3.6.1.6.3.10.1.1.2"/>
              <usmUserAuthPassword ValueType="OCTET_STRING" Value="Md5DesUser"/> ← 🔒
              <usmUserPrivProtocol ValueType="OID" Value="1.3.6.1.6.3.10.1.2.2"/>
              <usmUserPrivPassword ValueType="OCTET_STRING" Value="Md5DesUser"/> ← 🔒
              <usmUserStorageType ValueType="UINT32" Value="4"/>
              <usmUserStatus ValueType="UINT32" Value="1"/>
            </a>
            <b>
              <usmUserSecurityName ValueType="OCTET_STRING" Value="ShaDesUser"/>
              <usmUserAuthProtocol ValueType="OID" Value="1.3.6.1.6.3.10.1.1.3"/>
              <usmUserAuthPassword ValueType="OCTET_STRING" Value="ShaDesUser"/> ← 🔒
              <usmUserPrivProtocol ValueType="OID" Value="1.3.6.1.6.3.10.1.2.2"/>
              <usmUserPrivPassword ValueType="OCTET_STRING" Value="ShaDesUser"/> ← 🔒
              <usmUserStorageType ValueType="UINT32" Value="4"/>
              <usmUserStatus ValueType="UINT32" Value="1"/>
            </b>
          </usmUserTable>
        </usmUser>
      </snmpUsmMIB>
    <snmpVacmMIB>
      <vacmMIBObjects>
        <vacmMIBViews>
```

```

<vacmViewTreeFamilyTable>
  <a>
    <vacmViewTreeFamilyViewName ValueType="OCTET_STRING" Value="PublicView"/>
    <vacmViewTreeFamilySubtree ValueType="OID" Value="1"/>
    <vacmViewTreeFamilyMask ValueType="OCTET_STRING" Value="#128;"/>
    <vacmViewTreeFamilyType ValueType="UINT32" Value="1"/>
    <vacmViewTreeFamilyStorageType ValueType="UINT32" Value="4"/>
    <vacmViewTreeFamilyStatus ValueType="UINT32" Value="1"/>
  </a>
  <b>
    <vacmViewTreeFamilyViewName ValueType="OCTET_STRING" Value="PrivateView"/>
    <vacmViewTreeFamilySubtree ValueType="OID" Value="1"/>
    <vacmViewTreeFamilyMask ValueType="OCTET_STRING" Value="#128;"/>
    <vacmViewTreeFamilyType ValueType="UINT32" Value="1"/>
    <vacmViewTreeFamilyStorageType ValueType="UINT32" Value="4"/>
    <vacmViewTreeFamilyStatus ValueType="UINT32" Value="1"/>
  </b>
</vacmViewTreeFamilyTable>
</vacmMIBViews>
<vacmSecurityToGroupTable>
  <a>
    <vacmSecurityModel ValueType="UINT32" Value="1"/>
    <vacmSecurityName ValueType="OCTET_STRING" Value="PublicUser"/>
    <vacmGroupName ValueType="OCTET_STRING" Value="Snmpv1RWPublicGrp"/>
    <vacmSecurityToGroupStorageType ValueType="UINT32" Value="4"/>
    <vacmSecurityToGroupStatus ValueType="UINT32" Value="1"/>
  </a>
  <b>
    <vacmSecurityModel ValueType="UINT32" Value="1"/>
    <vacmSecurityName ValueType="OCTET_STRING" Value="PrivateUser"/>
    <vacmGroupName ValueType="OCTET_STRING" Value="Snmpv1RWPrivateGrp"/>
    <vacmSecurityToGroupStorageType ValueType="UINT32" Value="4"/>
    <vacmSecurityToGroupStatus ValueType="UINT32" Value="1"/>
  </b>
  <c>
    <vacmSecurityModel ValueType="UINT32" Value="2"/>
    <vacmSecurityName ValueType="OCTET_STRING" Value="PublicUser"/>
    <vacmGroupName ValueType="OCTET_STRING" Value="Snmpv2cRWPublicGrp"/>
    <vacmSecurityToGroupStorageType ValueType="UINT32" Value="4"/>
    <vacmSecurityToGroupStatus ValueType="UINT32" Value="1"/>
  </c>
  <d>
    <vacmSecurityModel ValueType="UINT32" Value="2"/>
    <vacmSecurityName ValueType="OCTET_STRING" Value="PrivateUser"/>
    <vacmGroupName ValueType="OCTET_STRING" Value="Snmpv2cRWPublicGrp"/>
    <vacmSecurityToGroupStorageType ValueType="UINT32" Value="4"/>
    <vacmSecurityToGroupStatus ValueType="UINT32" Value="1"/>
  </d>
  <e>
    <vacmSecurityModel ValueType="UINT32" Value="3"/>
    <vacmSecurityName ValueType="OCTET_STRING" Value="Md5DesUser"/>
    <vacmGroupName ValueType="OCTET_STRING" Value="AuthPrivGrp"/>
    <vacmSecurityToGroupStorageType ValueType="UINT32" Value="4"/>
    <vacmSecurityToGroupStatus ValueType="UINT32" Value="1"/>
  </e>
  <f>
    <vacmSecurityModel ValueType="UINT32" Value="3"/>
    <vacmSecurityName ValueType="OCTET_STRING" Value="ShaDesUser"/>
    <vacmGroupName ValueType="OCTET_STRING" Value="AuthPrivGrp"/>
    <vacmSecurityToGroupStorageType ValueType="UINT32" Value="4"/>
    <vacmSecurityToGroupStatus ValueType="UINT32" Value="1"/>
  </f>
  <g>
    <vacmSecurityModel ValueType="UINT32" Value="3"/>
    <vacmSecurityName ValueType="OCTET_STRING" Value="AuthNoPrivUser"/>
    <vacmGroupName ValueType="OCTET_STRING" Value="AuthNoPrivGrp"/>
    <vacmSecurityToGroupStorageType ValueType="UINT32" Value="4"/>
    <vacmSecurityToGroupStatus ValueType="UINT32" Value="1"/>
  </g>
  <h>
    <vacmSecurityModel ValueType="UINT32" Value="3"/>
    <vacmSecurityName ValueType="OCTET_STRING" Value="NoAuthNoPrivUser"/>
    <vacmGroupName ValueType="OCTET_STRING" Value="NoAuthNoPrivGrp"/>
    <vacmSecurityToGroupStorageType ValueType="UINT32" Value="4"/>
    <vacmSecurityToGroupStatus ValueType="UINT32" Value="1"/>
  </h>
</vacmSecurityToGroupTable>
<vacmAccessTable>
  <a>
    <vacmAccessContextPrefix ValueType="OCTET_STRING" Value="AuthPrivGrp"/>
    <vacmAccessSecurityModel ValueType="UINT32" Value="3"/>
    <vacmAccessSecurityLevel ValueType="UINT32" Value="3"/>
  </a>

```

```

<vacmAccessContextMatch ValueType="UINT32" Value="1"/>
<vacmAccessReadViewName ValueType="OCTET_STRING" Value="PublicView"/> ← Enable/Disable SNMPv3
<vacmAccessWriteViewName ValueType="OCTET_STRING" Value="PublicView"/> ← Enable/Disable SNMPv3
<vacmAccessNotifyViewName ValueType="OCTET_STRING" Value="PublicView"/>
<vacmAccessStorageType ValueType="UINT32" Value="4"/>
<vacmAccessStatus ValueType="UINT32" Value="1"/>
</a>
<b>
<vacmAccessContextPrefix ValueType="OCTET_STRING" Value="AuthNoPrivGrp"/>
<vacmAccessSecurityModel ValueType="UINT32" Value="3"/>
<vacmAccessSecurityLevel ValueType="UINT32" Value="2"/>
<vacmAccessContextMatch ValueType="UINT32" Value="1"/>
<vacmAccessReadViewName ValueType="OCTET_STRING" Value="PublicView"/> ← Enable/Disable SNMPv3
<vacmAccessWriteViewName ValueType="OCTET_STRING" Value="PublicView"/> ← Enable/Disable SNMPv3
<vacmAccessNotifyViewName ValueType="OCTET_STRING" Value="PublicView"/>
<vacmAccessStorageType ValueType="UINT32" Value="4"/>
<vacmAccessStatus ValueType="UINT32" Value="1"/>
</b>
<c>
<vacmAccessContextPrefix ValueType="OCTET_STRING" Value="NoAuthNoPrivGrp"/>
<vacmAccessSecurityModel ValueType="UINT32" Value="3"/>
<vacmAccessSecurityLevel ValueType="UINT32" Value="1"/>
<vacmAccessContextMatch ValueType="UINT32" Value="1"/>
<vacmAccessReadViewName ValueType="OCTET_STRING" Value="PublicView"/> ← Enable/Disable SNMPv3
<vacmAccessWriteViewName ValueType="OCTET_STRING" Value="PublicView"/> ← Enable/Disable SNMPv3
<vacmAccessNotifyViewName ValueType="OCTET_STRING" Value="PublicView"/>
<vacmAccessStorageType ValueType="UINT32" Value="4"/>
<vacmAccessStatus ValueType="UINT32" Value="1"/>
</c>
<d>
<vacmAccessContextPrefix ValueType="OCTET_STRING" Value="Snmpv1RWPUBLICGrp"/>
<vacmAccessSecurityModel ValueType="UINT32" Value="1"/>
<vacmAccessSecurityLevel ValueType="UINT32" Value="1"/>
<vacmAccessContextMatch ValueType="UINT32" Value="1"/>
<vacmAccessReadViewName ValueType="OCTET_STRING" Value="PublicView"/> ← Enable/Disable SNMPv1
<vacmAccessWriteViewName ValueType="OCTET_STRING" Value="PublicView"/> ← Enable/Disable SNMPv1
<vacmAccessNotifyViewName ValueType="OCTET_STRING" Value="PublicView"/>
<vacmAccessStorageType ValueType="UINT32" Value="4"/>
<vacmAccessStatus ValueType="UINT32" Value="1"/>
</d>
<e>
<vacmAccessContextPrefix ValueType="OCTET_STRING" Value="Snmpv1RWPrivateGrp"/>
<vacmAccessSecurityModel ValueType="UINT32" Value="1"/>
<vacmAccessSecurityLevel ValueType="UINT32" Value="1"/>
<vacmAccessContextMatch ValueType="UINT32" Value="1"/>
<vacmAccessReadViewName ValueType="OCTET_STRING" Value="PublicView"/> ← Enable/Disable SNMPv1
<vacmAccessWriteViewName ValueType="OCTET_STRING" Value="PublicView"/> ← Enable/Disable SNMPv1
<vacmAccessNotifyViewName ValueType="OCTET_STRING" Value="PrivateView"/>
<vacmAccessStorageType ValueType="UINT32" Value="4"/>
<vacmAccessStatus ValueType="UINT32" Value="1"/>
</e>
<f>
<vacmAccessContextPrefix ValueType="OCTET_STRING" Value="Snmpv2cRWPUBLICGrp"/>
<vacmAccessSecurityModel ValueType="UINT32" Value="2"/>
<vacmAccessSecurityLevel ValueType="UINT32" Value="1"/>
<vacmAccessContextMatch ValueType="UINT32" Value="1"/>
<vacmAccessReadViewName ValueType="OCTET_STRING" Value="PublicView"/> ← Enable/Disable SNMPv2
<vacmAccessWriteViewName ValueType="OCTET_STRING" Value="PublicView"/> ← Enable/Disable SNMPv2
<vacmAccessNotifyViewName ValueType="OCTET_STRING" Value="PublicView"/>
<vacmAccessStorageType ValueType="UINT32" Value="4"/>
<vacmAccessStatus ValueType="UINT32" Value="1"/>
</f>
<g>
<vacmAccessContextPrefix ValueType="OCTET_STRING" Value="Snmpv2cRWPrivateGrp"/>
<vacmAccessSecurityModel ValueType="UINT32" Value="2"/>
<vacmAccessSecurityLevel ValueType="UINT32" Value="1"/>
<vacmAccessContextMatch ValueType="UINT32" Value="1"/>
<vacmAccessReadViewName ValueType="OCTET_STRING" Value="PublicView"/> ← Enable/Disable SNMPv2
<vacmAccessWriteViewName ValueType="OCTET_STRING" Value="PublicView"/> ← Enable/Disable SNMPv2
<vacmAccessNotifyViewName ValueType="OCTET_STRING" Value="PrivateView"/>
<vacmAccessStorageType ValueType="UINT32" Value="4"/>
<vacmAccessStatus ValueType="UINT32" Value="1"/>
</g>
</vacmAccessTable>
</vacmMIBObjects>
</snmpVacmMIB>
</snmpModules>
</snmpV2>
</SnmpAgentConfig>

```

Enabling/Disabling SNMPv1, SNMPv2 and SNMPv3

By default, the parameters in the SNMP Agent section enable SNMPv1 and SNMPv2. However, you may want to disable them.

► To enable SNMPv1 and SNMPv2:

1. Ensure that the *Value* parameters of the fields `<vacmAccessReadViewName>` and `<vacmAccessWriteViewName>` are set to "PublicView" in the following groups:
 - Snmpv1RWPublicGrp
 - Snmpv1RWPrivateGrp
 - Snmpv2cRWPublicGrp
 - Snmpv2cRWPrivateGrp

These fields are identified in [Figure 69 on page 131](#) with the following icon:  Enable/Disable SNMPv1-2.

► To disable SNMPv1 and SNMPv2:

1. Ensure that the fields `<vacmAccessReadViewName>` and `<vacmAccessWriteViewName>` are empty in the following groups:
 - Snmpv1RWPublicGrp
 - Snmpv1RWPrivateGrp
 - Snmpv2cRWPublicGrp
 - Snmpv2cRWPrivateGrp

These fields are identified in [Figure 69 on page 131](#) with the following icon:  Enable/Disable SNMPv1-2.

► To enable SNMPv3:

1. Ensure that the *Value* parameters of the fields `<vacmAccessReadViewName>` and `<vacmAccessWriteViewName>` are set to "PublicView" in the following groups:
 - AuthPrivGrp
 - AuthNoPrivGrp
 - NoAuthNoPrivGrp

These fields are identified in [Figure 69 on page 131](#) with the following icon:

 Enable/Disable SNMPv3.

► To disable SNMPv3:

1. Ensure that the fields `<vacmAccessReadViewName>` and `<vacmAccessWriteViewName>` are empty in the following groups:
 - AuthPrivGrp
 - AuthNoPrivGrp
 - NoAuthNoPrivGrp

These fields are identified in [Figure 69 on page 131](#) with the following icon:

 Enable/Disable SNMPv3.

Changing SNMPv3 Credentials

The SNMP Agent section provides default Media5 credentials for SNMPv3. You can change these credentials.

► To change SNMPv3 credentials:

1. Change the password in the following fields:
 - usmUserAuthPassword (section **Md5DesUser**)
 - usmUserPrivPassword (section **Md5DesUser**)
 - usmUserAuthPassword (section **ShaDesUser**)
 - usmUserPrivPassword (section **ShaDesUser**)

These fields are identified in [Figure 69 on page 131](#) with the following icon : .



Caution: SNMPv3 passwords must be at least 8 characters long.

MIB Structure

The current MIB structure is defined in the SMI file, called *MX-SMI.my*. The SMI contains seven main groups.

Table 66: Structure of Management Information

Group	Description
mediatrixProducts	Each Media5 product has been assigned with its own sysObjectID value.
mediatrixAdmin	Root of the modules used for the administration of the products.
mediatrixMgmt	Root of the modules used to manage the products.
mediatrixConfig	Root of the modules used to configure the products.
mediatrixIpTelephony Signaling	Root of the modules used to configure the signalling protocols.
mediatrixModules	Provides a root in which modules can register their module entity. No MIB variables actually appear under this node.
mediatrixExperimental	The experimental sub-tree is the area where objects and events in MIBs under development can be placed without fear of conflicting with other MIBs. When the items rooted under an experimental sub-tree are ready for release, the sub-tree is re-attached under a permanent branch. Please note that Media5' configuration tool – the Unit Manager Network – does not support MIBs that are located under the <i>mediatrixExperimental</i> branch of the MIB structure. The Unit Manager Network does not have specific tasks to manage variables in experimental MIBs. Even though the Unit Manager Network can view experimental MIBs, SNMP operations may not work properly on them.

All parameters in the MIBs have been configured by default upon start up. However, if you need to modify some of these parameters (for example, parameters related to the country in which you are), use a MIB browser.

Textual Conventions

Textual conventions are defined in a module to ensure that all variables throughout the MIB structure use the same syntax and types. The type of each variable is defined in the *Composed syntax* line.

Table 67: Textual Conventions

Type	Definition
MxIpHostName	Represents an IP address or a domain name.
MxIpAddress	Represents an IP address.
MxIpPort	The TCP or UDP port number range. Values can be between 1 and 65535.
MxIpSubnetMask	Represents an Internet subnet mask.
MxIpSelect ConfigSource	Indicates the source to use during the next restart sequence for the provisioning of the localHost MIB objects. - static: uses static values provided by the user (such as DNS addresses, router, etc.). - dhcp: uses the DHCP server to retrieve the configuration of the localHost MIB objects.
MxIpConfigSource	Indicates the source used during the last restart sequence for the provisioning of the localHost MIB objects. - static: the user provided static values such as DNS addresses, router, etc. - dhcp: the DHCP server was used to retrieve the configuration of the localHost MIB objects. - Default: hardcoded values for recovery mode were used.
MxIpDhcpSite SpecificCode	Represents a DHCP site specific code. Values can be between 128 and 254 or 0. You can enter this code in your DHCP server to define IP addresses. Refer to "Chapter 8 - IP Address and Network Configuration" on page 143 for more details.
MxFloatingPoint	Represents a floating point number.
MxAdvancedIpPort	The TCP or UDP port number range. Values can be between 0 and 65535. The port number value 0 is used for special functionality defined in the variable definition.
MxEnableState	Represents an enabled/disabled state (boolean value).
MxActivationState	Represents an active/inactive state (boolean value).
MxSignalingAddress	Represents a valid signalling address.
MxDigitMap	A digit map is a sequence used to determine when the dialing of DTMFs is completed. See "Chapter 21 - Digit Maps" on page 293 for more details.

Objects, Conformance, and Events

Each MIB may have three types of data.

Table 68: MIB Data Types

Type	Description
Object	Represents the actual variables that can be set.
Conformance	Describes one or more groups to which the product may conform. This allows to have an exact idea of what a unit supports by glancing at the conformance information.
Event	An event is sent to tell what type of data will be received, but not the data itself. This is used to "warn" in advance what is coming.

IP Addresses

The MIB structure contains many IP addresses that can be set or viewed. See [“Chapter 8 - IP Address and Network Configuration” on page 143](#) for more details.

Persistence

A variable may either be persistent or volatile.

Table 69: Storage Clauses

Clause	Definition
Persistent	<i>Persistent</i> parameters are saved into the unit's memory and restored when it restarts. All the variables with the <i>Access = Read Write</i> attribute are persistent, except the variables representing commands (such as <i>sysAdminCommand</i>).
Volatile	<i>Volatile</i> parameters are lost every time the unit restarts. This type of parameter includes toggling parameters such as requesting a configuration file or a software download. <i>Statistics</i> are also volatile parameters that are lost every time the unit restarts.

Changing a Parameter Value

The Mediatrix 4100 software parameters are fully programmable by using the SNMP protocol. There are two ways to set up and configure a unit:

- ▶ By using a SNMP browser to contact the MIBs of the Mediatrix 4100. It is assumed that you have basic knowledge of TCP/IP network administration.
You can use the MIB browser built in the Media5' Unit Manager Network. See [“Unit Manager Network – Element Management System” on page xxii](#) for more details.
You can also use any third-party SNMP browser or network management application running the SNMP protocol to monitor and configure the Mediatrix 4100. However, the information may not be presented in the same manner depending on the SNMP browser used.
- ▶ By using the graphical user interface of the Management Server.
The Management Server could be Media5's Unit Manager Network. See [“Unit Manager Network – Element Management System” on page xxii](#) for more details.

Be sure to use the MIB files that match the version of the MIB located inside the current software build of the unit.

Locate the proper parameter to modify and change (SET) its value. Most of the parameters require to restart the Mediatrix 4100 unit. A restart may be software-initiated or manually initiated by unplugging the unit. It deletes all statistics stored and overwrites all volatile parameter values in the configuration file. A restart also reinitiates the entire unit's initial provisioning sequence.



Note: When performing a SET operation on any MIB variable, Media5 recommends to wait at least 30 seconds before shutting down the unit. This gives time to the software to update configuration data in flash memory.

Tables

There are two types of tables used in the MIB structure. They contain:

- ▶ Generic variables that apply to each line of a unit. This avoids to repeat each set of variables for each line it has.
- ▶ The administrative commands and status related to a managed object.

Generic Variables

All tables used to set variables for one or more lines (such as the *voiceIfTable*) are based on the *ifTable*, or interface table.

The *ifTable* lists the interfaces of a unit. In other words, it basically defines the lines that are used by the unit. It contains an *ifIndex*, which defines the interfaces. It may also contain interfaces such as:

- ▶ the LoopBack (*lo*) and Ethernet (*eth0*) interfaces.
- ▶ the actual voice interfaces (lines) of the unit.

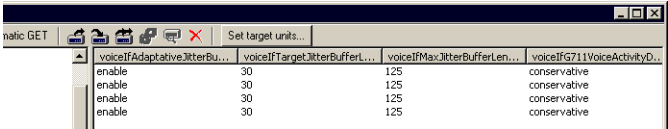
[Table 70](#) gives an example of the *ifTable*.

Table 70: ifTable Example

ifIndex	Type	Description
1	LoopBack	lo
2	Ethernet(0)	eth0
3	Voice FXS	(0)
4	Voice FXS	(1)
5	Voice FXS	(2)
...	...	...
26	Voice FXS	(24)

[Figure 70](#) shows a table built in the Unit Manager Network from the *voiceIfTable* parameters.

Figure 70: voiceIfTable Example



The screenshot shows a window titled 'natic GET' with a toolbar and a table. The table has four columns: 'voiceIfAdaptiveJitterBu...', 'voiceIfTargetJitterBufferL...', 'voiceIfMaxJitterBufferLen...', and 'voiceIfG711VoiceActivityD...'. The first column contains the value 'enable' for four rows. The second column contains the value '30' for four rows. The third column contains the value '125' for four rows. The fourth column contains the value 'conservative' for four rows.

voiceIfAdaptiveJitterBu...	voiceIfTargetJitterBufferL...	voiceIfMaxJitterBufferLen...	voiceIfG711VoiceActivityD...
enable	30	125	conservative
enable	30	125	conservative
enable	30	125	conservative
enable	30	125	conservative

You can perform GET and SET operations on these parameters.

Variables for Administrative Commands

Administrative commands are built on a hierarchical structure of parents-children. A command applied on a parent is propagated to all of its children.

There are two tables used to define administrative commands to groups:

- ▶ *groupAdmin*: A group may be the unit itself (gateway) or other instances. There are no instances other than the gateway defined at this moment.
- ▶ *ifAdmin*: This table applies to each interface of the unit.

groupAdmin Table

The *groupAdmin* table sends administrative commands at the highest instance in a hierarchy (such as the gateway).

Table 71: groupAdmin Parameters

Parameter	Description
groupSetAdmin	Command to set the administrative state of the system.
groupAdminState	The administrative state of the group. Indicates the current maintenance state of a group. Available states are unlocked, shutting down, and locked.
groupOpState	The operational state of the group. It reflects the group's internal state. Available states are enabled and disabled.
groupUsageState	The usage state of the group. Indicates the running state of a group. Available states are idle, active, busy, and idle-unusable.
groupAdminType	The type of resources managed by the group.
groupAdminDescription	The description of the group.
groupAdminParent Group	The parent's group. This is the index (<i>groupAdminIndex</i>), taken from this table (<i>groupAdminTable</i>), of the group that is the parent. If there is no parent, the value "-1" is used.

ifAdmin Table

The *ifAdmin* table is similar to the *groupAdmin* table, except that it applies to interfaces.

Table 72: ifAdmin Parameters

Parameter	Description
ifAdminSetAdmin	Command to set the administrative state of the current interface.
ifAdminAdminState	The administrative state of the current interface. It indicates the current maintenance state of a gateway component. Available states are unlocked, shutting down, locked, and permanentLock.
ifAdminOpState	The operational state of the current interface. This state reflects the component's internal state. Available states are enabled and disabled.
ifAdminUsageState	The usage state of the current interface. It indicates the running state of a voice component. Available states are idle, active, busy, and idle-unusable.
ifAdminParentType	The parents type of this interface.
ifAdminParent	The index of the parent of this interface.

SNMP Access Limitation

The SNMP access to the Mediatrix 4100 can be limited to only one of its interface or all interfaces.

► **To limit the access to the SNMP interface:**

1. In the *snmpAgentMIB*, select the interface where the Mediatrix 4100 can be accessed via SNMP in the *snmpAgentAccess* variable.

You have the following choices:

Table 73: SNMP Access Limitation Parameters

Access	Description
lanOnly	SNMP connections are only permitted on the LAN side, which is usually associated with the <i>ETH2</i> connector. The LAN IP address is provisioned by the <i>lanStaticAddress</i> variable.
wanOnly	SNMP connections are only permitted on the WAN side, which is usually associated with the <i>ETH1</i> connector. However, if the WAN interface is down and the unit reverts to its LAN configuration, the SNMP agent can access the Mediatrix 4100 on its LAN interface.
all	SNMP connections are permitted on both the LAN and WAN sides.

SNMP Service Agent

You can define whether the unit can be configured via SNMP or exclusively via its web interface.

1. In the *snmpAgentMIB*, define the SNMP service agent status in the *snmpAgentEnable* variable.
The SNMP service agent controls whether the Mediatrix 4100 may be accessed via SNMP or not.
 - enable: SNMP connections are allowed.
 - disable: SNMP connections are forbidden.

If you are exclusively using the web page to configure the Mediatrix 4100, you could disable the SNMP service agent and forbid any SNMP connection to the unit.

If the SNMP agent is disabled and the unit enters into recovery mode, SNMP connections are allowed during that time. Once the unit returns to its normal state, the SNMP service agent will revert to the configuration you have set.

Current MIB Version

You can find out the version of the MIB currently in the Mediatrix 4100.

1. In the *sysMgmtMIB*, locate the *sysMibVersion* variable.
This variable displays the current version of the MIB.

Sending Configuration Data to the Mediatrix 4100

The configuration data can be provisioned into the Mediatrix 4100 in two ways:

- ▶ as a configuration file sent from the Management Server to the Mediatrix 4100 via TFTP
- ▶ as a MIB sent from the Management Server to the Mediatrix 4100 via SNMP

Configuration File

The configuration file is the fastest way to deliver the necessary information. This may be important when initializing a large number of units at the same time. The configuration file is mostly used for the initial provisioning sequence (see [“Initial Provisioning Sequence” on page 14](#) for more details).

For more information on how to use a configuration file for updating the Mediatrix 4100, see [“Chapter 12 - Configuration File Download” on page 187](#).

Management Information Base – MIB

Sending information via SNMP means that individual variables can be changed without sending the whole MIB. You could use a dual system where a configuration file is sent for initial configuration and a MIB browser / SNMP browser is used to implement minor changes.

The Mediatrix 4100 has several configurable MIBs. All variables in these MIBs have been configured by default upon start up. However, if you need to modify some of these variables, use a MIB browser.

IP Address and Network Configuration

The Mediatrix 4100 must be provisioned with various IP addresses and network parameters to be fully functional. This occurs each time the Mediatrix 4100 is started or when an IP address value is changed in the MIB. The Mediatrix 4100 can use static network parameters as well as parameters provided by a DHCP server or even a DNS.

This chapter assumes that you know how to set up and use a DHCP and DNS server. If not, ask your network administrator to set up DHCP-related variables.

This chapter also refers to the MIB structure of the configuration variables. Refer to [“Chapter 7 - MIB Structure and SNMP” on page 127](#) for more details.

IP Addresses

The MIB structure contains IP addresses that can be set or viewed. These IP addresses are physically located in their relevant MIB. For instance, the IP addresses for the Syslog daemon are located in the *syslogMIB*. However, when viewing the MIB structure in a MIB browser such as the Media5 Unit Manager Network, the IP addresses are grouped in two distinct folders for easy management.

Table 74: IP Addresses Folders

Folder	Description
ipAddressStatus	Lists all the IP addresses used by the unit, in read-only format.
ipAddressConfig	Lists all the IP addresses you can set. Changes made in this folder are reflected in the <i>ipAddressStatus</i> folder.

IP Addresses Formats in the DHCP Server

You can use a number of formats when defining IP addresses in the DHCP server.

Table 75: IP Addresses Formats in DHCP Server

Format	Description	Allowed Char.
Decimal	You can enter IP addresses in the widely-used (base 10) decimal format. For instance, a decimal IP address would be 192.168.0.9. IP addresses cannot contain decimal numbers higher than 255.	0..9
Hexadecimal	You can enter IP addresses in (base 16) hexadecimal format. Prepending “0x” to the value instructs the unit to interpret it as hexadecimal. For instance, the decimal IP address 192.168.0.9 translates to 0xC0.0xA8.0x0.0x9 in hexadecimal format.	0..9, A..F
Octal	You can enter IP addresses in (base 8) octal format. Prepending “0” to the value instructs the unit to interpret it as octal. For instance, the decimal IP address 192.168.0.9 translates to 0300.0250.00.011 in octal format.	0..7

You can make combinations of the three bases in a single string, because each number in the string is interpreted separately. For instance, 0300.0xA8.000.9 translates to the decimal IP address 192.168.0.9.

There may be some confusion between the three available IP address formats. In particular, it is important to understand that prefixing "0" to the values makes them interpreted as octal values. For instance, the string 192.168.0.009 is not valid because 009 is interpreted in octal, and the digit "9" does not exist in that base.

Provisioning Source

The Mediatrix 4100 IP information may come from a variety of sources.

Table 76: IP Address Provisioning Sources

Source	Description
Static	You manually enter the value and it remains the same every time the Mediatrix 4100 restarts. If you do not specify a value, a default static value applies.
DHCP	The value is obtained at start-time by querying a DHCP server and using standard DHCP fields or options. See RFC 2131 section 2 and RFC 2132.
DHCP – Site specific options	The value is obtained at start-time by querying a DHCP server and using a non-standard option specific to the site where the Mediatrix 4100 is used. See "Site Specific Options" on page 157 for more details.
DHCP – Vendor specific options	The value is obtained at start-time by querying a DHCP server and using a standard option that is reserved for storing vendor specific information. See "Vendor Specific Options" on page 156 for more details.
DNS	The value is obtained at start-time by querying a DNS server.
None	The value is not provisioned. The application provides an acceptable default.

Services

This section describes the services the Mediatrix 4100 uses and their settings. Most of these services require that you define their IP address and, if required, port number. See ["DHCP Server Configuration" on page 155](#) for more details.

Configuration variables of network parameters are defined in the MIB structure under the *ipAddressConfig* folder. This folder is subdivided into groups, one for each service that requires a network parameter.

Configuration Source

The configuration your Mediatrix 4100 uses can either be:

- ▶ dynamically assigned (network parameters assigned by a DHCP Server)
- ▶ static (network parameters you manually defined in the MIB structure)

You can also set these parameters via the web interface, as described in ["Network Settings" on page 35](#).

DHCP Configuration

Using DHCP-assigned IP addresses ensures that the Mediatrix 4100 receives the addresses that are stored in the DHCP server. This assumes that you have previously set the DHCP server with the proper values. See ["DHCP Server Configuration" on page 155](#) for more details.

The Mediatrix 4100 can receive numerous information from the DHCP server, including the vendor or site specific information. Note that the Mediatrix 4100 does not make a DHCP request in the following cases:

- ▶ If all MIB variables *xxSelectConfigSource* are set to **static** at start-up.
- ▶ If one of the MIB variables *xxSelectConfigSource* is set to **dhcp** after the initialization process.

Verifying the DHCP-Assigned IP Addresses

You can query the MIB structure to see the IP addresses that have been assigned to the Mediatrix 4100. Those IP addresses are located under the *ipAddressStatus* folder in read-only variables.

This assumes that you know the local host IP address. There are two ways to get the local host IP address of a Mediatrix unit:

- ▶ Connect a telephone into one of the FXS ports of the Mediatrix unit, dial “*#*0” and listen for the IP address that is given.
- ▶ Use the autodetect feature of the Media5 Unit Manager Network product. See [“Unit Manager Network – Element Management System” on page xxii](#) for more details.

Static Configuration

Using static IP addresses allows you to bypass the DHCP server or still be able to use the Mediatrix 4100 if you are not running a DHCP server.

In this case, having one or more configuration source variable set to DHCP slows down the restart process. If any information is set to come from the DHCP server (for example, SNTP address), the restarting unit waits for a maximum period of two minutes if the DHCP server cannot be reached, even if most other settings are set to “static”.

The reason for this delay is that the Mediatrix 4100 cannot function as configured if part of its configuration (the DHCP information) is unavailable. To avoid this problem, you can set all configuration sources the Mediatrix 4100 supports to “static”.



In the *Unit Manager Network Administration Manual*, refer to chapter *Performing Actions on Mediatrix Units*, section *Removing all DHCP Options*.

▶ To set all configuration sources to static:

1. In the *sysAdminMIB*, set the *sysAdminCommand* variable to *setConfigSourcesStatic*.

Local Host

The *ipAddressConfigLocalHost* group allows you to set the IP information the Mediatrix 4100 needs to work properly. This group is vital to the proper operation of the Mediatrix 4100. If a variable of this group is not properly set, the Mediatrix 4100 may not be able to restart and be contacted after it has restarted.



In the *Unit Manager Network Administration Manual*, refer to chapter *Administration Parameters*, section *IP Configuration*.

▶ To select the local host configuration source:

1. In the *ipAddressConfig* folder, locate the *localHostSelectConfigSource* variable (under the *ipAdressConfigLocalHost* group).
2. Set this variable to either **static** or **dhcp**.

Table 77: Local Host Variables

Variable	Default Static Value	DHCP Source
localHostAddress	“192.168.0.1”	Yiaddr field
localHostPrimaryDns ^a	“192.168.0.10”	Option 6 (first of the list)
localHostSecondaryDns ^a	“192.168.0.10”	Option 6 (second of the list)
localHostDefaultRouter ^b	“192.168.0.10”	Option 3 (first of the list)
localHostSubnetMask	“255.255.255.0”	Option 1
localHostDhcpServer	“” (cannot be set)	Siaddr field

a. If you do not want to use a DNS, set the variable to 0.

- b. If you are not using a default router, set the variable to **0.0.0.0**. Setting the default router IP address to "0.0.0.0" may lead to software download problems. See the troubleshooting section ["Software Upgrade Issues" on page 358](#) for more details.



Note: Media5 recommends not to set a static subnet mask address of 255.255.255.254 because this would only create a subnet with two addresses. This only leaves one bit host addresses. Since a subnet must have a network (all bits 0) and a broadcast address (all bits 1), this leaves no room for hosts.



Note: If the *localHostDnsOverrideEnable* variable is enabled, the primary and secondary DNS addresses are set with static values. See ["Static DNS" on page 147](#) for more details.

In the table above, the only variables that allow an empty string are: *localHostPrimaryDns*, *localHostSecondaryDns* and *localHostDefaultRouter*.

3. Restart the Mediatrix 4100 so that the changes may take effect.

WAN Address Configuration Source

The Wide Area Network (WAN) address is the public IP address attributed to the Mediatrix 4100. This address is used for incoming signalling, media and management traffic.

► To set the WAN IP address configuration source:

1. In the *ipAddressConfig* folder, locate the *localHostWanAddressSelectConfigSource* variable (under the *ipAddressConfigLocalHost* group).
This variable indicates the source to be used for the provisioning of the WAN address. It offers the following choices:

Table 78: WAN IP Address Source Settings

Option ^a	Description
localAddress	The WAN address is the one that is set in the <i>localHostAddress</i> variable, whereas the <i>localHostStaticWanAddress</i> is ignored.
static	The Mediatrix 4100 has a static WAN address. The address is configured in the <i>localHostStaticWanAddress</i> variable. Note that this setting allows a limited NAT traversal scheme.

a. ppoe is not available on the Mediatrix 4100.

Table 79: WAN IP Address Source

Variable	Default Static Value	DHCP Source
localHostWanAddress	"192.168.0.1"	Option IP-Address

2. Restart the Mediatrix 4100 so that the changes may take effect.

SNMP Configuration

No DHCP value is available, you can define SNMP information with only static values.

Table 80: SNMP Source

Variable	Default Static Value	DHCP Source
localhostSnmpPort	161	N/A



In the *Unit Manager Network Administration Manual*, refer to chapter *Working with SNMP*, section *Setting Unit SNMP Preferences*.

The Mediatrix 4100 uses the SNMP protocol for software configuration. Set the following SNMP-related variable to properly use the protocol.

Table 81: SNMP Configuration Variables

Variable	Description
localhostStaticSnmpPort	Default SNMP agent port, which is the port number to use to reach the local host via SNMP protocol. Restart the unit to update this parameter. Default Value: 161 Note: If you change the SNMP agent port, change the port used in the management server or MIB Browser. Not doing so will prevent you from contacting the unit. The Management Server could be the Media5 Unit Manager Network. See “Unit Manager Network – Element Management System” on page xxii for more details.

You can query the SNMP information assigned by the DHCP server in the following variables (in the *ipAddressStatus* folder):

- ▶ localhostSnmpPort
- ▶ msTrapPort

Static DNS

By default, the Mediatrix 4100 receives DNS IP addresses according to the configuration source you have defined in the *localhostSelectConfigSource* variable. In general, these addresses are provided by an ISP (Internet Service Provider) via DHCP.

However, you can set static values for the primary and secondary DNS IP addresses, even when the Mediatrix 4100 is set by DHCP. These static values can thus override DHCP provisioning

The Mediatrix 4100 may receive DNS addresses from two sources:

- ▶ from an ISP via DHCP
- ▶ from the static local host DNS IP addresses

[Table 82](#) explains how DNS addresses are attributed to the Mediatrix 4100.

Table 82: DNS Addresses Possibilities

Configuration Source	localhostDns OverrideEnable	DNS address of Mediatrix 4100
DHCP	disabled	DHCP DNS
DHCP	enabled	static local host DNS

► To use static DNS IP addresses:

1. In the *ipAddressConfig* folder, set the *localHostDnsOverrideEnable* variable (under the *ipAdressConfigLocalHost* group) to **enable**.
The primary DNS and secondary DNS addresses are set with the static values defined in the *localHostStaticPrimaryDns* and *localHostStaticSecondaryDns* variables.
2. Restart the Mediatrix 4100 so that the changes may take effect.

Image

The *ipAddressConfigImage* group provides the configuration necessary to download applications into the Mediatrix 4100. This includes emergency downloads in case of repetitive failure to start the main application.



In the *Unit Manager Network Administration Manual*, refer to chapter *Administration Parameters*, section *Software and Emergency Download*.

► To select the Image configuration source:

1. In the *ipAddressConfig* folder, locate the *imageSelectConfigSource* variable (under the *ipAddressConfigImage* group).
2. Set this variable to either **static** or **dhcp** (vendor/site specific option).

Table 83: Image Information Source

Variable	Default Static Value	DHCP Source
imagePrimaryHost	"192.168.0.10"	Use option specified in variable <i>imageDhcpPrimarySiteSpecificCode</i> , bytes 0-3. If not specified (0), use option 43, sub-option 117, bytes 0-3.
imagePrimaryPort	69 ^a	Use option specified in variable <i>imageDhcpPrimarySiteSpecificCode</i> , bytes 4-5. If not specified (0), use option 43, sub-option 117, bytes 4-5. If bytes 4-5 are not present, use the default static value.
imageSecondaryHost	"192.168.0.10"	Use option specified in variable <i>imageDhcpSecondarySiteSpecificCode</i> , bytes 0-3. If not specified (0), use option 43, sub-option 118, bytes 0-3.
imageSecondaryPort	69 ^a	Use option specified in variable <i>imageDhcpSecondarySiteSpecificCode</i> , bytes 4-5. If not specified (0), use option 43, sub-option 118, bytes 4-5. If bytes 4-5 are not present, use the default static value.

a. This is the well-known TFTP port number as per RFC 1340.

Management Server

The *ipAddressConfigMs* group provides the configuration necessary for contacting a SNMP management server such as the Media5 Unit Manager Network.



In the *Unit Manager Network Administration Manual*, refer to chapter *Administration Parameters*, section *Unit Manager Server*.

► To select the Management Server configuration source:

1. In the *ipAddressConfig* folder, locate the *msSelectConfigSource* variable (under the *ipAddressConfigMs* group).
2. Set this variable to either **static** or **dhcp** (vendor/site specific option).

Table 84: Management Server Source

Variable	Default Static Value	DHCP Source
msHost	N/A	Use option specified in variable <i>msDhcpSiteSpecificCode</i> , bytes 0-3. If not specified (0), use option 43, sub-option 200, bytes 0-3.
msStaticHost	"192.168.0.10"	N/A
msTrapPort	N/A	Use option specified in variable <i>msDhcpSiteSpecificCode</i> , bytes 4-5. If not specified (0), use option 43, sub-option 200, bytes 4-5. If bytes 4-5 are not present, use the default static value.
msStaticPort	162	N/A
msStaticTrapPort	162	N/A



Note: If you change the value of the *msStaticTrapPort* variable, change the port used in the management server. Not doing so will prevent you from viewing the received traps from the unit.

Configuration File Fetching

The *ipAddressConfigFileFetching* group provides the configuration necessary to contact the configuration file server when fetching a configuration file.

► **To select the configuration file fetching server configuration source:**

1. In the *ipAddressConfig* folder, locate the *configFileFetchingSelectConfigSource* variable (under the *ipAddressConfigFileFetching* group).
2. Set this variable to either **static** or **dhcp** (vendor/site specific option).

Table 85: Configuration File Fetching Source

Variable	Default Static Value	DHCP Source
configFileFetching Host	N/A	Use option specified in variable <i>configFileFetchingDhcpSiteSpecificCode</i> , bytes 0-3. If not specified (0), use option 43, sub-option 201, bytes 0-3.
configFileFetching Port	N/A	Use option specified in variable <i>configFileFetchingDhcpSiteSpecificCode</i> , bytes 4-5. If not specified (0), use option 43, sub-option 201, bytes 4-5. If bytes 4-5 are not present, use the default static value.
configFileFetching StaticHost	"192.168.0.10"	N/A
configFileFetching Static Port	69	N/A

3. Restart the Mediatrix 4100 so that the changes may take effect.

Syslog

The *ipAddressConfigSyslog* group provides the configuration necessary for contacting a Syslog server.



In the *Unit Manager Network Administration Manual*, refer to chapter *Administration Parameters*, section *Syslog Daemon*.

► **To select the Syslog configuration source:**

1. In the *ipAddressConfig* folder, locate the *syslogSelectConfigSource* variable (under the *ipAddressConfigSyslog* group).
2. Set this variable to either **static** or **dhcp**.

Table 86: Syslog Source

Variable	Default Static Value	DHCP Source
syslogHost	"192.168.0.10"	Use option specified in variable <i>syslogDhcpSiteSpecificCode</i> , bytes 0-3. If not specified (0), use option 43, sub-option 110, bytes 0-3.
syslogPort	514 ^a	Not provided by the DHCP, use the default static value.

a. The port number is as per RFC 1340.

SIP Servers

The *ipAddressConfigSipServer* group provides the configuration necessary for contacting different SIP servers.



In the *Unit Manager Network Administration Manual*, refer to chapter *Signalling Protocols Parameters*, section *SIP Configuration Window*.



Note: Although the DHCP option #120 is reserved for SIP servers, no standard currently defines the content and layout of this option.



Note: If, for a given server, the port is 0, then the host and port for this server are obtained through a DNS SRV request. See ["Chapter 10 - DNS SRV Configuration" on page 173](#) for more details.

► **To select the SIP Servers configuration source:**

1. In the *ipAddressConfig* folder, locate the *sipServerSelectConfigSource* variable (under the *ipAddressConfigSipServer* group).
2. Set this variable to either **static** or **dhcp** (vendor/site specific option).

Table 87: SIP Servers Source

Variable	Default Static Value	DHCP Source
sipHomeDomain ProxyHost	"192.168.0.10"	Use option specified in variable <i>sipHomeDomainProxyDhcpSiteSpecificCode</i> , bytes 0-3. If not specified (0), use option 43, sub-option 204, bytes 0-3
sipHomeDomain ProxyPort	0	Use option specified in variable <i>sipHomeDomainProxyDhcpSiteSpecificCode</i> , bytes 4-5. If not specified (0), use option 43, sub-option 204, bytes 4-5. If bytes 4-5 are not present, use the default static value.
sipOutboundProxy Host	"0.0.0.0"	Use option specified in variable <i>sipOutboundProxyDhcpSiteSpecificCode</i> , bytes 0-3. If not specified (0), use option 43, sub-option 205, bytes 0-3.
sipOutboundProxy Port	0	Use option specified in variable <i>sipOutboundProxyDhcpSiteSpecificCode</i> , bytes 4-5. If not specified (0), use option 43, sub-option 205, bytes 4-5. If bytes 4-5 are not present, use the default static value.
sipRegistrarHost	"192.168.0.10"	Use option specified in variable <i>sipRegistrarDhcpSiteSpecificCode</i> , bytes 0-3. If not specified (0), use option 43, sub-option 203, bytes 0-3.
sipRegistrarPort	0	Use option specified in variable <i>sipRegistrarDhcpSiteSpecificCode</i> , bytes 4-5. If not specified (0), use option 43, sub-option 203, bytes 4-5. If bytes 4-5 are not present, use the default static value.

Table 87: SIP Servers Source (Continued)

Variable	Default Static Value	DHCP Source
sipPresenceCompositorHost	"0.0.0.0"	Use option specified in variable <i>sipPresenceCompositorDhcpSiteSpecificCode</i> , bytes 0-3. If not specified (0), use option 43, sub-option 206, bytes 0-3.
sipPresenceCompositorPort	0	Use option specified in variable <i>sipPresenceCompositorDhcpSiteSpecificCode</i> , bytes 4-5. If not specified (0), use option 43, sub-option 206, bytes 4-5. If bytes 4-5 are not present, use the default static value.

SNTP

The *ipAddressConfigSntp* group provides the configuration necessary for contacting a NTP/SNTP server.

If you are using a NTP or SNTP server (see ["Chapter 20 - SNTP Settings" on page 289](#) for more details), the DHCP server already has options that can be set to provide time server addresses, and the order in which clients use them to attempt to discover servers.

The Mediatrix 4100 uses *Option 42* to specify the IP address corresponding to the server that provides NTP/ SNTP (RFC 1769).



In the *Unit Manager Network Administration Manual*, refer to chapter *Administration Parameters*, section *SNTP*.

► **To select the SNTP configuration source:**

1. In the *ipAddressConfig* folder, locate the *sntpSelectConfigSource* variable (under the *ipAddressConfigSntp* group).
2. Set this variable to either **static** or **dhcp**.

Table 88: SNTP Source

Variable	Default Static Value	DHCP Source
sntpHost	"192.168.0.10"	Option 42 (first of the list).
sntpPort	123	Not provided by the DHCP, use the default static value.

ETH2 Connector Static IP Address

You can use the *ETH2* connector of the Mediatrix 4100 with the network card of a computer. You could then use this computer to directly access the unit via its LAN interface.

This section describes how to set the static IP address of the *ETH2* connector and use this LAN static IP address according to the VLAN substitution or always enabled configuration.

► **To set the *ETH2* connector static IP address:**

1. In the *ipRoutingMIB*, set the *lanStaticAddressActivation* variable to one of the following values:

Table 89: LAN Connector Static IP Address

Parameter	Description
ipRouting	To set the <i>lanStaticAddress</i> variable as the Local Area Network (LAN) IP address used by the Mediatrix 4100's <i>ETH2</i> interface, set the variable <i>ipRoutingEnable</i> to enable .
always	Sets the <i>lanStaticAddress</i> as the Local Area Network (LAN) IP address used by the Mediatrix 4100's <i>ETH2</i> interface.

2. In the *ipAddressConfig* folder, set the static *ETH2* connector information as follows:

Table 90: LAN Interface Source

Variable	Default Static Value	Description
lanStaticAddress	192.168.10.1	LAN IP address used by the unit's <i>ETH2</i> interface.
lanStaticNetworkMask	255.255.255.0	LAN subnet mask used by the unit's <i>ETH2</i> interface.



Note: Do not set the *lanStaticAddress* variable to 0.0.0.0. This could prevent the unit from properly sending a DHCP discover request.



Note: Media5 recommends not to set the *lanStaticNetworkMask* variable to 255.255.255.254 because this would only create a subnet with two addresses. This only leaves one bit host addresses. Since a subnet must have a network (all bits 0) and a broadcast address (all bits 1), this leaves no room for hosts.

3. Restart the Mediatrix 4100 so that the changes may take effect.

DHCP Configuration

The following sections describes paramaters that you can set on the Mediatrix 4100 to better interact with a DHCP server.

DHCP Options Waiting Time

Many network switches use the Spanning Tree Protocol (STP) to manage Ethernet ports activity. If the Mediatrix 4100 is connected to such a switch, the switch may shut down the matching Ethernet port for at least one minute. This shutdown on the switch side can prevent proper communication. It may thus take up to three minutes before the unit receives an answer to its request for DHCP options.

You can configure the Mediatrix 4100 to bypass this waiting period by restarting without a valid address for the requested servers.

► **To configure how much time the Mediatrix 4100 waits for DHCP options:**

1. In the *sysConfigMIB*, set the amount of time (in seconds) the Mediatrix 4100 will wait for DHCP options in the *sysConfigDhcpWaitDelay* variable.
This variable is only effective when the variable *sysConfigDhcpWait* is set to **disable**. The default value is **10** seconds.
2. Define the restart behaviour of a unit that needs to issue a DHCP request to receive some options in the *sysConfigDhcpWait* variable.
This variable has no influence on the wait behaviour in effect when the unit is requesting an IP address lease from the DHCP server.

Table 91: DHCP Wait Parameters

Parameter	Description
enable	The unit waits up to 130 seconds for DHCP options and then, in the absence of a response, proceeds to restart without a valid address for the requested servers. This is the default value.
disable	The unit restarts, without valid server addresses, if a DHCP response has not been received after the amount of time specified in the variable <i>sysConfigDhcpWaitDelay</i> .

Bootp BROADCAST Flag in DHCP Requests

Standards Supported	• RFC 1542 – Clarifications and Extensions for the Bootstrap Protocol
----------------------------	---

You can define whether the Mediatrix 4100 sets the Bootp BROADCAST flag in DHCP requests it issues. This applies to the DHCP Discover requests the unit sends on startup. It does not apply to the DHCP Discover requests sent when renewing the lease.

► **To define the BROADCAST flag behaviour:**

1. In the *sysConfigMIB*, define the BROADCAST flag behaviour in the *sysConfigBootpFlags* variable.

Table 92: BROADCAST Flag Parameters

Parameter	Description
noFlags	The Bootp BROADCAST flag is not set in DHCP requests that the Mediatrix 4100 issues. The DHCP server may thus answer by using unicast delivery.
broadcastFlag	The Bootp BROADCAST flag is set in DHCP requests that the Mediatrix 4100 issues.

Changing the Size of DHCP Requests

You can append a string to the value used as Vendor Class ID (Option 60) in a DHCP request. This option is useful when servers require that DHCP packets sent to them be of a minimum size. A string of arbitrary characters (including blanks) can then be used to artificially increase the size of DHCP requests.

See [“Vendor Class ID” on page 157](#) for more details.

► **To change the size of DHCP requests:**

1. In the *sysConfigMIB*, define the string to append in the *sysConfigProductNamePadding* variable.

DHCP Server Configuration

Standards Supported

- RFC 2131 – Dynamic Host Configuration Protocol, section 2
- RFC 2132 – DHCP Options and BOOTP Vendor Extensions

DHCP servers generally allocate a range of IP addresses for use on a network and reserve IP addresses for specific devices using a unique identifier for each device. The Mediatrix 4100 unique identifier is its media access control (MAC) address.



Note: Media5 recommends to use a Windows 2000- or Unix-based DHCP server. If you run Windows NT 4.0 and use the built-in Microsoft DHCP Server, use the Site Specific instead of Vendor Specific information.

You can locate the MAC address as follows:

- ▶ on the label located on the bottom side of the unit.
- ▶ in the *sysMgmtMIB* under the *sysMacAddress* variable.
- ▶ you can dial the following digits on a telephone connected to the Mediatrix 4100:
*#*1

The Mediatrix 4100 answers back with its MAC address. See [“Special Vocal Features” on page 15](#) for more details.

Media5 recommends to reserve an IP address with an infinite lease for each Mediatrix 4100 on the network.

Connection to the DHCP Behaviour

When the Mediatrix 4100 restarts, it requests a DHCP offer to get its IP addresses and network information. The Mediatrix 4100 waits four seconds before sending another request. The delay between each request is increased exponentially after each request up to a maximum delay of 64 seconds, and then restarts at a 4 seconds delay.

- ▶ first request: 4 seconds delay
- ▶ second request: 8 seconds delay
- ▶ third request: 16 seconds delay
- ▶ fourth request: 32 seconds delay
- ▶ fifth request: 64 seconds delay
- ▶ sixth request: 4 seconds delay
- ▶ seventh request: 8 seconds delay
- ▶ etc.

The Mediatrix 4100 stops broadcasting as soon as it receives at least one reply. If the offer is valid, the Mediatrix 4100 takes it and continues its initialization procedure.



Note: If the *localhostSelectConfigSource* variable is set to **static** and any other *xxSelectConfigSource* variable is set to **dhcp**, the Mediatrix 4100 makes its DHCP request that will be released immediately.

Network Configuration

[Table 93](#) lists some of the network options to configure in the DHCP server:

Table 93: Network Configuration

Information	Description	Option	Data Format	Example
Subnet Mask	Specifies subnet configuration	001	xxx.xxx.xxx.xxx	255.255.255.0
Routers	List of routers on your network	003	Array of IP Addresses	192.168.10.1 192.168.10.2

Table 93: Network Configuration (Continued)

Information	Description	Option	Data Format	Example
DNS Servers	List of DNS servers on your network	006	Array of IP Addresses	192.168.10.11 192.168.10.12

Vendor and Site Specific DHCP Options

This section briefly describes vendor and site specific DHCP options.

Most of the MIB variables described in [“Services” on page 144](#) require that you define their IP address and, if required, port number. When defining these variables, you can do so in two ways: via vendor specific options or site specific options.

The default value is to use the vendor specific codes. In this case, the *xxSiteSpecificCode* MIB variables are set to 0.

If you want to use site specific codes instead, change the value of the *xxSiteSpecificCode* MIB variables from the default value (0) to the value you select in the DHCP server. See [“Site specific code definition” on page 160](#) for an example of vendor specific and site specific settings.

Vendor Specific Options

Standards Supported	RFC 2132 – DHCP Options and BOOTP Vendor Extensions, section 8.4 (“Vendor-specific options”)
----------------------------	--

The vendor specific DHCP option is a standard DHCP option used to store information specific to the vendor of the DHCP client. The vendor specific option code is 43. Because there are different information elements that can be stored in this option, each element has been allocated a “sub-option” number. See [Table 94 on page 157](#) for the complete list.

Like all other options, the vendor specific information field (option 43) first contains a code (43), a length (in byte) and some data that spans the number of bytes specified in the length.

The data is organized as a series of sub-options, each of them laid-out like a regular option (code, length, data). The codes can be anything between 1 and 254, and the vendor, Media5, chooses these codes. See [Table 94 on page 157](#) for actual codes.

The following figures show the general and encapsulated layout of the vendor specific information option.

Figure 71: General Layout of a Vendor Specific Information Option

43	Len	Data	Data	Data	Data	...
----	-----	------	------	------	------	-----

Figure 72: Layout for Encapsulated Vendor Specific Options

43	Len	Code1	Len1	Data1	Data1	...	Code2	Len2	Data2	Data2	...
----	-----	-------	------	-------	-------	-----	-------	------	-------	-------	-----

[Figure 73](#) is an example of a vendor specific option containing an *mshost* IP address (192.168.1.2).

Figure 73: Example of Encapsulated Vendor Specific Option

43	6	200	4	192	168	1	2
----	---	-----	---	-----	-----	---	---

Mediatix units store two types of information in vendor specific options: IP addresses with optional port number and FQDNs with optional port number. The layout for storing IP addresses is explained in section [“Entering IP Addresses” on page 158](#). The layout for storing FQDNs is explained in section [“Entering FQDNs” on page 159](#).

Vendor Class ID

When using the vendor specific option, first define a Vendor Class ID for the Mediatrix 4100 (not supported in Windows NT servers). A Vendor Class ID can be used by DHCP clients to identify their vendor type and configuration. When using this option, vendors can define their own specific identifier values to convey a particular hardware or operating system configuration or other identifying information.

Where vendor classes are used, the DHCP server responds to identifying clients by using option code 43, the reserved option type for returning vendor specific information to the client.

DHCP servers that do not interpret this option type are expected to ignore it when it is specified by clients.

Please refer to your DHCP server's documentation to learn how to create a new vendor class.



Note: The class to add is *Mediatrix 4108*, *Mediatrix 4116*, or *Mediatrix 4124*, depending on your unit model.

Creating Vendor Specific Information

Once the Vendor ID Class is created, place the proper values in the 43 option of the DHCP server. The 43 option contains sub-options that are encapsulated (according to the format described in RFC 2132).

If the option is not in the DHCP server, the Mediatrix 4100 uses an invalid value (0.0.0.0:0).

Please refer to your DHCP server's documentation to learn how to create vendor specific information. See ["Entering IP Addresses" on page 158](#) for more details on the syntax to use.

Site Specific Options

Standards Supported

RFC 2132 – DHCP Options and BOOTP Vendor Extensions, section 2 ("BOOTP Extension/DHCP Option Field Format").

Site specific options are non-standard DHCP options specific to the network where the Mediatrix 4100 is used. You are responsible to allocate an option number (between 128 and 254) for each information element to be stored.

Mediatrix units store two types of information in site specific options: IP addresses with optional port number and FQDNs with optional port number. The layout for storing IP addresses is explained in section ["Entering IP Addresses" on page 158](#). The layout for storing FQDNs is explained in section ["Entering FQDNs" on page 159](#). [Figure 74](#) is an example of site specific option #146, containing address 192.168.0.1.

Figure 74: Site Specific Option Example

146	4	192	168	0	1
-----	---	-----	-----	---	---

When using the site specific option, you can place the values in the site specific options of your choice in the DHCP server. You must then enter the values in the proper MIB variables.

Please refer to your DHCP server's documentation to learn how to create site specific information. See ["Entering IP Addresses" on page 158](#) for more details on the syntax to use.

Option Codes

This table lists all vendor specific sub-option codes.

Table 94: Sub-Option Codes

Code		Description
Decimal	Hexadec.	
110	0x6E	Syslog Server address and port.

Table 94: Sub-Option Codes (Continued)

Code		Description
Decimal	Hexadec.	
117	0x75	Image Primary Server host address and port. The default port number is 69 if you are using TFTP as protocol. The default port number is 80 if you are using HTTP as protocol.
118	0x76	Image Secondary Server host address and port. The default port number is 69 if you are using TFTP as protocol. The default port number is 80 if you are using HTTP as protocol.
200	0xC8	Management Server SNMP Trap host address and port.
201	0xC9	Configuration file fetching host address and port. The default port number is 69 if you are using TFTP as protocol. The default port number is 80 if you are using HTTP as protocol.
203	0xCB	SIP Registrar host address and port.
204	0xCC	SIP Home Domain Proxy host address and port.
205	0xCD	SIP Outbound Proxy host address and port.
206	0xCE	SIP Presence Compositor host address and port.

Entering IP Addresses

In the DHCP server, IP addresses can be entered in decimal, hexadecimal or octal format. See [“IP Addresses” on page 143](#) for more details.

There are two formats of address string:

- ▶ Long: Has a size of 6 bytes (12 hexadecimal characters) and includes the IP address and port.
- ▶ Short: Has a size of 4 bytes (8 hexadecimal characters) and includes only the IP address. In this case, the default port is used.

Numeric values are stored in network byte order (Big-Endian).

Table 95: Address String Formats

Variable	Valid Range	Typical Value	Note
IP Address	Any valid IP address	192.168.0.2 (hex. 0xC0.0xA8.0x0.0x2)	N/A
Port	1 - 32,768	162 (hex. 0xA2)	Not present in the format with dimension 4.

When entering IP addresses in the DHCP server, there is a difference between the vendor specific option and the site specific option.

The vendor specific options must be encapsulated because more than one information can be stored in this option:

```
[code][length][4-6 bytes address][another code][another length][another address]...
```

The site specific options can have only one information per option:

```
[4-6 bytes address]
```

The DHCP server adds the proper code and length in the packet it sends out.

Example

The following example shows how to enter the Syslog (code 110) IP address 192.168.0.10 (with the default port used) and the same address at port 2545 in hexadecimal format.

Figure 75: Example – Short Address String

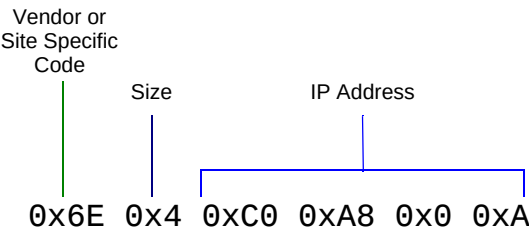
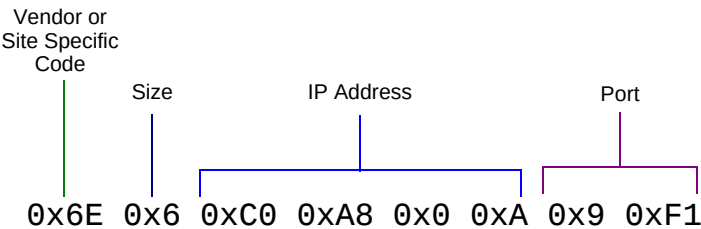


Figure 76: Example – Long Address String



Entering FQDNs

The FQDN address layout is a Media5 proprietary extension to the IP address layout. This format allows the configuration of an IP address in binary format (with or without port) or a FQDN in string format (with or without port) in the same option. The method to decode the information is based on the length of the option: a length of 4 or 6 is decoded as an IP address in binary format and a length higher than 6 is decoded as a FQDN in string format.

The IP address in binary format (with and without port) is explained in section [“Entering IP Addresses” on page 158](#).

The FQDN in string format consists of an array of characters representing the FQDN address.

Figure 77: FQDN String Format (without a port number)

Code	Len (7 to n)	FQDN char 1	FQDN char 1	...	FQDN char n
------	--------------	-------------	-------------	-----	-------------

You can specify a port by adding the port number in string format after a ':' at the end of the FQDN.

Figure 78: FQDN String Format (without a port number)

Code	Len (7 to n)	FQDN char 1	FQDN char 1	...	FQDN char n	":" (0x3A)	Port char 1	...	Port char y
------	--------------	-------------	-------------	-----	-------------	------------	-------------	-----	-------------

The space or null (ASCII code 0) character can be used as padding at the end of the string to have a length higher than 6, since all spaces and nulls are ignored. Note that an IP address can be defined in string format.

Examples

The following are some examples of the DHCP server configuration (based on linux dhcpd).

Vendor specific options – option vendor-encapsulated-options

- ▶ Syslog Server (IP address "192.168.0.1" in binary format).
6e:04:c0:a8:00:01:

- ▶ Primary Image Server (IP address and port "192.168.0.10:6000" in binary format).
75:06:c0:a8:00:01:17:70:
- ▶ Secondary Image Server (IP address "192.168.0.1" in string format).
76:0b:31:39:32:2e:31:36:38:2e:30:2e:31:
- ▶ Management Server (IP address and port "192.168.0.1:6000" in string format).
c8:10:31:39:32:2e:31:36:38:2e:30:2e:31:3a:36:30:30:30:
- ▶ Configuration File Fetching (FQDN "server.com").
c9:0a:73:65:72:76:65:72:2e:63:6f:6d:
- ▶ SIP Registrar Server (FQDN and port "server.com:6000").
cb:0f:73:65:72:76:65:72:2e:63:6f:6d:3a:36:30:30:30:
- ▶ SIP Proxy Server (FQDN with space padding "svr").
cc:07:73:76:72:20:20:20:20:
- ▶ SIP Outbound Proxy Server (FQDN and port with space padding "svr:12").
cd:07:73:76:72:3a:31:32:20:
- ▶ SIP Presence Server (FQDN with a null terminated string "server.com").
ce:0b:73:65:72:76:65:72:2e:63:6f:6d:00;

Site specific options

- ▶ Syslog Server (IP address "192.168.0.1" in binary format). The IP-address or string format of dhcpd can be used.
option mx-syslog-ip 192.168.0.1;
Or "option mx-syslog-str c0:a8:00:01;"
- ▶ Primary Image Server (IP address and port "192.168.0.10:6000" in binary format).
option mx-primary-image-str c0:a8:00:01:17:70;
- ▶ Secondary Image Server (IP address "192.168.0.1" in string format).
option mx-secondary-image-str "192.168.0.1";
- ▶ Management Server (IP address and port "192.168.0.1:6000" in #string format).
option mx-ms-str "192.168.0.1:6000";
- ▶ Configuration File Fetching (FQDN "server.com").
option mx-filefetching-str "server.com";
- ▶ SIP Registrar Server (FQDN and port "server.com:6000").
option mx-sip-registrar-str "server.com:6000";
- ▶ SIP Proxy Server (FQDN with space padding "svr").
option mx-sip-proxy-str "svr " ;
- ▶ SIP Outbound Proxy Server (FQDN and port with space padding "svr:12").
option mx-sip-outbound-proxy-str "svr:12 " ;
- ▶ SIP Presence Server (FQDN with a null terminated string "server.com"). The FQDN is expressed in hexadecimal to be able to put a null character.
option mx-sip-presence-proxy-str 73:65:72:76:65:72:2e:63:6f:6d:00;

Site specific code definition

```
option mx-syslog-ip code 129 = ip-address;
option mx-syslog-str code 129 = string;
option mx-primary-image-ip code 130 = ip-address;
option mx-primary-image-str code 130 = string;
option mx-secondary-image-ip code 131 = ip-address;
option mx-secondary-image-str code 131 = string;
option mx-ms-ip code 132 = ip-address;
option mx-ms-str code 132 = string;
option mx-filefetching-ip code 133 = ip-address;
option mx-filefetching-str code 133 = string;
option mx-sip-registrar-ip code 134 = ip-address;
option mx-sip-registrar-str code 134 = string;
option mx-sip-proxy-ip code 135 = ip-address;
option mx-sip-proxy-str code 135 = string;
```

```
option mx-sip-outbound-proxy-ip code 136 = ip-address;
option mx-sip-outbound-proxy-str code 136 = string;
option mx-sip-presence-ip code 137 = ip-address;
option mx-sip-presence-str code 137 = string;
```

Settings Example

Let's say for instance you want:

- ▶ the Image server at 10.3.2.154 (static)
- ▶ the Management Server via DHCP in the vendor specific options
- ▶ the Syslog server via DHCP in the site specific option #250

The following are the corresponding MIB values:

- ▶ imageSelectConfigSource = static
- ▶ imageStaticPrimaryHost = 10.3.2.154
- ▶ msSelectConfigSource = dhcp
- ▶ msDhcpSiteSpecificCode = 0
- ▶ syslogSelectConfigSource = dhcp
- ▶ syslogDhcpSiteSpecificCode = 250

The following is the corresponding DHCP setup, assuming the Management server is located at 10.3.2.201 and the Syslog server is located at 10.3.2.200 (port 1024):

- ▶ Option 43 (vendor specific option) contains the hexadecimal sequence 0xC80x40xA0x30x20xC9 **inserted among other sequences**.

Table 96: Hexadecimal Sequence - Option 43

Hexadecimal Part	Corresponding Information
0xC8	code 200 (management server)
0x4	size of 4 bytes
0xA0x30x20xC9	IP address 10.3.2.201

- ▶ Option 250 (site specific option) contains the hexadecimal sequence 0xA0x30x20xC80x400.

Table 97: Hexadecimal Sequence - Option 250

Hexadecimal Part	Corresponding Information
0xA0x30x20xC8	IP address 10.3.2.200
0x400	port 1024

Error Handling

In the event of a network or server failure, this section describes the application behaviour and/or replacement values to use.

Table 98: Replacement Values for Error Recovery

Type	Variable	Replacement value
IP address	(All variables of that type)	0.0.0.0
String	(All variables of that type)	""

DHCP Server Failures

If the Mediatrix 4100 cannot contact the DHCP server, it performs one of the following actions:

1. Retries contacting the DHCP server until it answers. The Mediatrix 4100 does not restart.
2. Uses the replacement value from [Table 98](#) for all variables that depend on the DHCP.

This assumes that the Mediatrix 4100 is set to get its IP information via a DHCP server.

If the Mediatrix 4100 is configured to request some DHCP options but does not require an IP address from the server, the amount of time it will wait for an answer before proceeding further is configurable as defined in ["Appendix - DHCP Options Waiting Time" on page 153](#).

Vendor/Site Specific Option Missing

If a vendor specific or site specific option is missing from the DHCP server answer, the Mediatrix 4100 uses the replacement value from [Table 98](#) for each variable that depends on missing vendor/site specific options.

DNS Failures

If the DNS cannot be contacted, the Mediatrix 4100 performs the following steps:

1. The Mediatrix 4100 sends a first request to the primary DNS server.
2. If the DNS server cannot be contacted within two seconds, the Mediatrix 4100 sends a request to the secondary DNS server.
3. If the secondary DNS server cannot be contacted, the Mediatrix 4100 uses the replacement value from [Table 98](#) for all variables that depend on the DNS.

Ethernet Connection Speed

You can set the speed of the Ethernet connection of the Mediatrix 4100.

You can also set these parameters via the web interface, as described in ["Ethernet Connection Speed" on page 35](#).

► To set the Ethernet connection speed:

1. In the *sysConfigMIB*, set the Ethernet connection speed of the:
 - *ETH1* connector in the *sysConfigNetworkEthernetSpeed* variable
 - *ETH2* connector in the *sysConfigComputerEthernetSpeed* variable.

The following values are available:

- Auto detect
- 10Mbs-HalfDuplex
- 100Mbs-HalfDuplex
- 10Mbs-FullDuplex
- 100Mbs-FullDuplex

A half-duplex connection refers to a transmission using two separate channels for transmission and reception, while a full-duplex connection refers to a transmission using the same channel for both transmission and reception.

If unknown, set the variable to **Auto detect** so that the Mediatrix 4100 can automatically detect the network speed.



Caution: Whenever you force a connection speed / duplex mode, be sure that the other device and all other intermediary nodes used in the communication between the two devices have the same configuration. See ["Speed and Duplex Detection Issues" on page 163](#) for more details.

Speed and Duplex Detection Issues

There are two protocols for detecting the Ethernet link speed:

- ▶ An older protocol called parallel detection.
- ▶ A more recent protocol called auto-negotiation (IEEE 802.3u).

The auto-negotiation protocol allows to detect the connection speed and duplex mode. It exchanges capabilities and establishes the most efficient connection. When both endpoints support the auto-negotiation, there are no problems. However, when only one endpoint supports auto-negotiation, the parallel detection protocol is used. This protocol can only detect the connection speed; the duplex mode cannot be detected. In this case, the connection may not be established.

The Mediatrix 4100 has the possibility to force the desired Ethernet link speed and duplex mode by disabling the auto-negotiation and selecting the proper setting (*sysConfigNetworkEthernetSpeed* variable). When forcing a link speed at one end, be sure that the other end (a hub, switch, etc.) has the same configuration. To avoid any problem, the link speed and duplex mode of the other endpoint must be exactly the same.

The Mediatrix 4100 uses the following types of servers:

- ▶ Registrar server
- ▶ Proxy server
- ▶ Outbound Proxy server
- ▶ Presence Compositor server

This chapter describes how to configure the Mediatrix 4100 to properly use these servers.

You can also set these parameters via the web interface, as described in [“SIP Servers Configuration” on page 63](#).

Registrar Server

The registrar server accepts REGISTER requests and places the information it receives in those requests into the location service for the domain it handles.



In the *Unit Manager Network Administration Manual*, refer to chapter *Signalling Protocols Parameters*, section *SIP Configuration Window*.

Configuration Source

The Mediatrix 4100 must know the IP address and port number of the Registrar server. You can assign these information to the Mediatrix 4100 through a DHCP server or manually enter them yourself with the static variables.



Note: When defining whether or not the Mediatrix 4100 must get its SIP server configuration through a DHCP server, this is set for all the SIP servers. You cannot define a different configuration for each type of server.

DHCP Configuration

Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See [“Chapter 8 - IP Address and Network Configuration” on page 143](#) for more details.

▶ To use DHCP-assigned information:

1. In the *ipAddressConfig* folder, locate the *sipServerSelectConfigSource* variable (under the *ipAddressConfigSipServer* group).
This variable defines whether or not the Mediatrix 4100 must get its registrar server configuration through a DHCP server.
2. Set the *sipServerSelectConfigSource* variable to **dhcp**.
You can query the registrar server's IP address and port number assigned by the DHCP server in the following read-only variables (under the *ipAddressStatusSipServer* group of the *ipAddressStatus* folder):
 - sipRegistrarHost
 - sipRegistrarPort

3. Set how you want to define the registrar server information in the DHCP server.

Table 99: Registrar Server DHCP Information

To use a...	Set...
vendor specific code	The <i>sipRegistrarDhcpSiteSpecificCode</i> variable (under the <i>ipAddressConfigSipServer</i> group) to 0 . Set the registrar server IP address in the DHCP server inside the vendor specific sub-option 203 (hexadecimal 0xCB).
site specific code	The <i>sipRegistrarDhcpSiteSpecificCode</i> variable to any value between 128 and 254. Set the registrar server IP address in the DHCP server inside the site specific option you have chosen (it must match the value of the <i>sipRegistrarDhcpSiteSpecificCode</i> variable in the unit's configuration).

Static Configuration

Use the static configuration if you are not using a DHCP server or if you want to bypass it.

► To use static information:

1. In the *ipAddressConfig* folder, locate the *sipServerSelectConfigSource* variable (under the *ipAddressConfigSipServer* group).
This variable defines whether or not the Mediatix 4100 must get its registrar server configuration through a DHCP server.
2. Set the *sipServerSelectConfigSource* variable to **static**.
3. Set the following variables:

Table 100: Registrar Server Static Information

Variable	Description
<i>sipRegistrarStaticHost</i>	Registrar server static IP address or domain name. Default Value: 192.168.0.10
<i>sipRegistrarStaticPort</i>	Registrar server static IP port number. Note: If this variable corresponds to a domain name that is bound to a SRV record, the port must be set to 0 for the unit to perform DNS requests of type SRV (as per RFC 3263). Otherwise, the unit will not use DNS SRV requests, but will rather use only requests of type A because it does not need to be specified which port to use. Default Value: 0

Proxy Server

The proxy server is an intermediary program that acts as both a server and a client for the purpose of making requests on behalf of other clients. A proxy server primarily plays the role of routing, which means its job is to ensure that a request is passed on to another entity that can further process the request. Proxies are also useful for enforcing policy and for firewall traversal. A proxy interprets, and, if necessary, rewrites parts of a request message before forwarding it.



In the *Unit Manager Network Administration Manual*, refer to chapter *Signalling Protocols Parameters*, section *SIP Configuration Window*.

Configuration Source

The Mediatrix 4100 must know the IP address and port number of the proxy server. You can assign these information to the Mediatrix 4100 through a DHCP server or manually enter them yourself with the static variables.



Note: When defining whether or not the Mediatrix 4100 must get its SIP server configuration through a DHCP server, this is set for all the SIP servers. You cannot define a different configuration for each type of server.

DHCP Configuration

Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See ["Chapter 8 - IP Address and Network Configuration" on page 143](#) for more details.

► To use DHCP-assigned information:

1. In the *ipAddressConfig* folder, locate the *sipServerSelectConfigSource* variable (under the *ipAddressConfigSipServer* group).
This variable defines whether or not the Mediatrix 4100 must get its proxy server configuration through a DHCP server.
2. Set the *sipServerSelectConfigSource* variable to **dhcp**.
You can query the proxy server's IP address and port number assigned by the DHCP server in the following read-only variables (under the *ipAddressStatusSipServer* group of the *ipAddressStatus* folder):
 - *sipHomeDomainProxyHost*
 - *sipHomeDomainProxyPort*
3. Set how you want to define the proxy server information in the DHCP server.

Table 101: Proxy Server DHCP Information

To use a...	Set...
vendor specific code	The <i>sipHomeDomainProxyDhcpSiteSpecificCode</i> variable (under the <i>ipAddressConfigSipServer</i> group) to 0 . Set the proxy server IP address in the DHCP server inside the vendor specific sub-option 204 (hexadecimal 0xCC).
site specific code	The <i>sipHomeDomainProxyDhcpSiteSpecificCode</i> variable (under the <i>ipAddressConfigSipServer</i> group) to any value between 128 and 254. Set the proxy server IP address in the DHCP server inside the site specific option you have chosen (it must match the value of the <i>sipHomeDomainProxyDhcpSiteSpecificCode</i> variable in the unit's configuration).

Static Configuration

Use the static configuration if you are not using a DHCP server or if you want to bypass it.

► To use static information:

1. In the *ipAddressConfig* folder, locate the *sipServerSelectConfigSource* variable (under the *ipAddressConfigSipServer* group).
This variable defines whether or not the Mediatix 4100 must get its proxy server configuration through a DHCP server.
2. Set the *sipServerSelectConfigSource* variable to **static**.
3. Set the following variables:

Table 102: Proxy Server Static Information

Variable	Description
sipHomeDomainProxyStatic Host	Proxy server static IP address or domain name. Default Value: 192.168.0.10
sipHomeDomainProxyStatic Port	Proxy server static IP port number. Note: If this variable corresponds to a domain name that is bound to a SRV record, the port must be set to 0 for the unit to perform DNS requests of type SRV (as per RFC 3263). Otherwise, the unit will not use DNS SRV requests, but will rather use only requests of type A because it does not need to be specified which port to use. Default Value: 0

Outbound Proxy Server

An outbound proxy is an intermediary entity that acts as both a server and a client for the purpose of making requests on behalf of other clients. The outbound proxy receives all outbound traffic and forwards it. Incoming traffic may or may not go through the outbound proxy. The outbound proxy's address is never used in the SIP packets, it is only used as a physical network destination for the packets.

When the outbound proxy is enabled, the proxy is still used to create the *To* and the *From* headers, but the packets are physically sent to the outbound proxy.

The outbound proxy is enabled if the IP address is valid (i.e., not 0.0.0.0). The default static value in the MIB is 0.0.0.0.

SIP Outbound Proxy (From RFC 3261)

A proxy that receives requests from a client, even though it may not be the server resolved by the Request-URI. Typically, a user agent is manually configured with an outbound proxy, or can learn about one through auto-configuration protocols.

When enabled, the initial route for all SIP requests contains the outbound proxy address, suffixed with the loose routing parameter "lr". The Request-URI still contains the home domain proxy address. Requests are directed to the first route (the outbound proxy).

Configuration Source

The Mediatrix 4100 must know the IP address and port number of the outbound proxy. You can assign these information to the Mediatrix 4100 through a DHCP server or manually enter them yourself with the static variables.



Note: When defining whether or not the Mediatrix 4100 must get its SIP server configuration through a DHCP server, this is set for all the SIP servers. You cannot define a different configuration for each type of server.

DHCP Configuration

Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See [“Chapter 8 - IP Address and Network Configuration” on page 143](#) for more details.

► To use DHCP-assigned information:

1. In the *ipAddressConfig* folder, locate the *sipServerSelectConfigSource* variable (under the *ipAddressConfigSipServer* group).
This variable defines whether or not the Mediatrix 4100 must ask for its outbound proxy settings through a DHCP server.
2. Set the *sipServerSelectConfigSource* variable to **dhcp**.
You can query the outbound proxy's IP address and port number assigned by the DHCP server in the following read-only variables (under the *ipAddressStatusSipServer* group of the *ipAddressStatus* folder):
 - *sipOutboundProxyHost*
 - *sipOutboundProxyPort*
3. Set how you want to define the outbound proxy server information in the DHCP server.

Table 103: Outbound Proxy Server DHCP Information

To use a...	Set...
vendor specific code	The <i>sipOutboundProxyDhcpSiteSpecificCode</i> variable (under the <i>ipAddressConfigSipServer</i> group) to 0 . Set the outbound proxy server IP address in the DHCP server inside the vendor specific sub-option 205 (hexadecimal 0xCD).
site specific code	The <i>sipOutboundProxyDhcpSiteSpecificCode</i> variable (under the <i>ipAddressConfigSipServer</i> group) to any value between 128 and 254. Set the outbound proxy server IP address in the DHCP server inside the site specific option you have chosen (it must match the value of the <i>sipOutboundProxyDhcpSiteSpecificCode</i> variable in the unit's configuration).

Static Configuration

Use the static configuration if you are not using a DHCP server or if you want to bypass it.

► To use static information:

1. In the *ipAddressConfig* folder, locate the *sipServerSelectConfigSource* variable (under the *ipAddressConfigSipServer* group).
This variable defines whether or not the Mediatrix 4100 must ask for its outbound proxy settings through a DHCP server.
2. Set the *sipServerSelectConfigSource* variable to **static**.

- Set the following variables:

Table 104: Outbound Proxy Static Information

Variable	Description
sipOutboundProxyStaticHost	Static outbound proxy server IP address or domain name. Default Value: 192.168.0.10
sipOutboundProxyStaticPort	Static outbound proxy server IP port number. Note: If this variable corresponds to a domain name that is bound to a SRV record, the port must be set to 0 for the unit to perform DNS requests of type SRV (as per RFC 3263). Otherwise, the unit will not use DNS SRV requests, but will rather use only requests of type A because it does not need to be specified which port to use. Default Value: 0

► **To disable the outbound proxy:**

- In the *ipAddressConfig* folder, set the *sipOutboundProxyStaticHost* variable to **0.0.0.0**.
To re-enable the outbound proxy, enter a valid IP address.
You can now specify if the outbound proxy uses a loose routing or strict routing type.

Loose Router Configuration

Standards Supported

RFC 3261 – SIP: Session Initiation Protocol, section 6
RFC 2543 – SIP: Session Initiation Protocol

You must specify the type of routing of the outbound proxy configured in *sipOutboundProxyHost* does.



Note: This feature is currently located under the *mediatrixExperimental* branch of the MIB structure. See [“MIB Structure” on page 135](#) for more details.

► **To set the outbound proxy router status:**

- In the *sipMIB*, locate the *sipOutboundProxyConfig* variable.
The following values are available:

Table 105: Outbound Proxy Router Status

Value	Description
looseRouter	This is the most current method for SIP routing, as per RFC 3261, and will become the standard behaviour once RFC 3261 compliance is achieved. See “SIP Outbound Proxy (From RFC 3261)” on page 168 for details.

Loose Router

A proxy is said to be loose routing if it follows the procedures defined in the *RFC 3261* specification (section 6) for processing of the *Route* header field. These procedures separate the destination of the request (present in the Request-URI) from the set of proxies that need to be visited along the way (present in the *Route* header field). A proxy compliant to these mechanisms is also known as a loose router.

Table 105: Outbound Proxy Router Status (Continued)

Value	Description
strictRouter	Pre-RFC 3261, RFC 2543 compatible SIP routing. The initial route for all SIP requests contains the home domain proxy address (the Request-URI). Requests are directed to the outbound proxy. In other words, the Request-URI is constructed as usual, using the home domain proxy and the user name, but is used in the route set. The Request-URI is filled by the outbound proxy address.

Presence Compositor Server

Standards Supported

- RFC 3863 – Presence Information Data Format (PIDF)
- RFC 3903 – Session Initiation Protocol (SIP) Extension for Event State Publication

The Presence Compositor server is a User Agent Server (UAS) that processes PUBLISH requests and is responsible for compositing event state into a complete, composite event state of a resource for a presentity. The presence Compositor is enabled if the IP address is valid (i.e., not 0.0.0.0). The default static value in the MIB is 0.0.0.0.

Configuration Source

The Mediatrix 4100 must know the IP address and port number of the Presence Compositor server. You can assign these information to the Mediatrix 4100 through a DHCP server or manually enter them yourself with the static variables.



Note: When defining whether or not the Mediatrix 4100 must get its SIP server configuration through a DHCP server, this is set for all the SIP servers. You cannot define a different configuration for each type of server.

DHCP Configuration

Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See [“Chapter 8 - IP Address and Network Configuration” on page 143](#) for more details.

► To use DHCP-assigned information:

1. In the *ipAddressConfig* folder, locate the *sipServerSelectConfigSource* variable (under the *ipAddressConfigSipServer* group).
This variable defines whether or not the Mediatrix 4100 must get its Presence Compositor server configuration through a DHCP server.
2. Set the *sipServerSelectConfigSource* variable to **dhcp**.
You can query the Presence Compositor server's IP address and port number assigned by the DHCP server in the following read-only variables (under the *ipAddressStatusSipServer* group of the *ipAddressStatus* folder):
 - sipPresenceCompositorHost
 - sipPresenceCompositorPort

3. Set how you want to define the Presence Compositor server information in the DHCP server.

Table 106: Presence Compositor Server DHCP Information

To use a...	Set...
vendor specific code	The <i>sipPresenceCompositorDhcpSiteSpecificCode</i> variable (under the <i>ipAddressConfigSipServer</i> group) to 0 . Set the proxy server IP address in the DHCP server inside the vendor specific sub-option 206 (hexadecimal 0xCE).
site specific code	The <i>sipPresenceCompositorDhcpSiteSpecificCode</i> variable (under the <i>ipAddressConfigSipServer</i> group) to any value between 128 and 254. Set the Presence Compositor server IP address in the DHCP server inside the site specific option you have chosen (it must match the value of the <i>sipPresenceCompositorDhcpSiteSpecificCode</i> variable in the unit's configuration).

Static Configuration

Use the static configuration if you are not using a DHCP server or if you want to bypass it.

► To use static information:

1. In the *ipAddressConfig* folder, locate the *sipServerSelectConfigSource* variable (under the *ipAddressConfigSipServer* group).
This variable defines whether or not the Mediatrix 4100 must get its Presence Compositor server configuration through a DHCP server.
2. Set the *sipServerSelectConfigSource* variable to **static**.
3. Set the following variables:

Table 107: Presence Compositor Server Static Information

Variable	Description
<i>sipPresenceCompositorStaticHost</i>	Presence Compositor server static IP address or domain name. Default Value: 0.0.0.0
<i>sipPresenceCompositorStaticPort</i>	Presence Compositor server static IP port number. Note: If this variable corresponds to a domain name that is bound to a SRV record, the port must be set to 0 for the unit to perform DNS requests of type SRV (as per RFC 3263). Otherwise, the unit will not use DNS SRV requests, but will rather use only requests of type A because it does not need to be specified which port to use. Default Value: 0

DNS SRV Configuration

This chapter describes the configuration required for the Mediatrix 4100 to work with a DNS SRV.

What is a DNS SRV?

Standards Supported

- RFC 2782 – A DNS RR for specifying the location of services (DNS SRV)
- RFC 3263 - Session Initiation Protocol (SIP): Locating SIP Servers

Currently, one must either know the exact address of a server to contact it, or broadcast a question.

DNS SRV is an extension of the standard DNS server. SRV (Service Record) is a type of entry a network administrator may put into the DNS answers. A DNS SRV is used to get one or more IP addresses of servers, each one having its own weight and priority.

Each server received when using DNS SRV, depending on its weight and priority, can be used as a primary or backup server or can be part of a load balancing system.

For instance, the client requests the SRV for SIP servers in some domain. The DNS server may return the A, B, and C addresses, which are all SIP servers. Each address has a weight and the client must choose one of those three addresses by using a random algorithm that considers the weight.

To use DNS SRV, an administrator must set a service records (SRV) into the DNS servers available on the network.

DNS SRV implementation should imply a shared database between servers since a REGISTER and an INVITE can be sent to any server, not necessarily the same one.

DNS SRV applies to both TCP and UDP transport types.

Priority vs Weight

A DNS SRV uses the *priority* and *weight* concepts to distribute the requests.

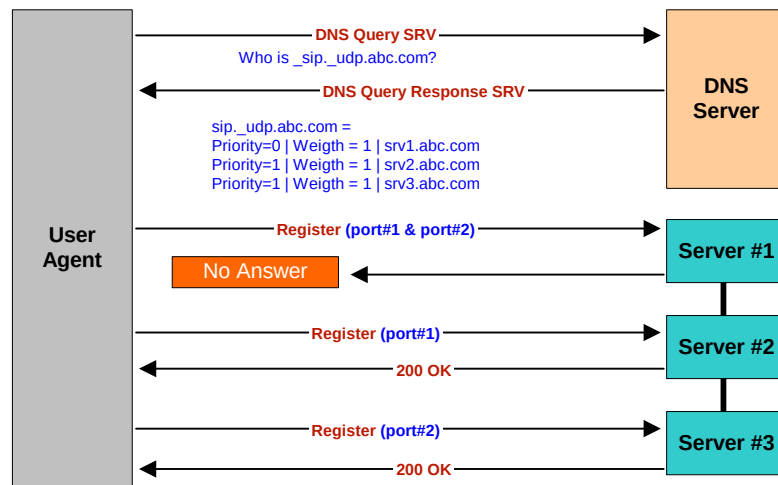
Table 108: Priority vs. Weight

Parameter	Description
Priority	A client must attempt to contact the target host with the lowest-numbered priority it can reach.
Weight	Specifies a relative weight for entries with the same priority. Larger weights should be given a proportionately higher probability of being selected.

DNS SRV Call Flow

The following is a standard DNS SRV call flow:

Figure 79: DNS SRV Call Flow



Enabling DNS SRV on the Mediatrix 4100

If the address of a service corresponds to a domain name that is bound to a SRV record, the port this service uses must be set to **0** for the unit to perform DNS requests of type SRV (as per RFC 3263). Otherwise, the unit will not use DNS SRV requests, but will rather use only requests of type A because it does not need to be specified which port to use. See [“Chapter 9 - SIP Servers” on page 165](#) for more details.

► To enable DNS SRV:

1. In the *ipAddressConfig* folder, locate the *sipServerSelectConfigSource* variable (under the *ipAddressConfigSipServer* group).
This variable defines whether the Mediatrix 4100 must get its proxy server configuration through a DHCP server or not.
2. Set the *sipServerSelectConfigSource* variable to **static**.
3. Set one or more of the following variables to **0**:

Table 109: Variables to Enable DNS SRV

Server	Variable to enable
SIP Registrar server	sipRegistrarStaticPort
SIP Proxy server	sipHomeDomainProxyStaticPort
SIP Outbound Proxy server	sipOutboundProxyStaticPort



Note: Any “SRV enabled” service must have a host name recognized by the DNS SRV server. “_sip._udp” or “_sip._tcp” (depending on the transport type) is automatically added to the host name.

DNS SRV Record Lock

You can configure the Mediatrix 4100 to always use the same DNS SRV record for a SIP call ID. As a result, a call or registration always uses the same destination until the destination is unreachable or the unit receives a different DNS SRV result.

► **To enable the DNS SRV record lock feature:**

1. In the *sipInteropMIB*, set the *sipInteropLockDnsSrvRecordPerCallEnable* variable to **enable**.
All messages during a call or registration use the same SRV record.
If you set this variable to **disable** (which is the default value), the Mediatrix 4100 rather follows the behaviour as described in RFC 3263.
2. Restart the Mediatrix 4100 so that the changes may take effect.

DNS SRV-Oriented Settings

The following parameters have an effect on the DNS SRV behaviour.

Table 110: DNS SRV-Oriented Settings

Parameter	Description
<i>sipInteropTransmissionTimeout</i>	• Has a dramatic effect should a server time out, since a default 32 s delay would be introduced at every call. • Media5 recommends a maximum of 2-3 s when using DNS SRV. • See “Transmission Timeout” on page 273 for more details.
<i>sipPenaltyBoxTime</i>	If <i>sipPenaltyBoxEnable</i> is set to enable : • A “timed out” server is considered “not responding” for this amount of time. • Can be seen as the time it will take to retry a server that failed responding. • See “SIP Penalty Box” on page 267 for more details.
<i>sipInteropReuseCredentialEnable</i>	If <i>sipInteropReuseCredentialEnable</i> is set to enable : • If there is not a shared database between servers, this could lead to authentication problems because a REGISTER and an INVITE can be sent to any server, not necessarily the same one. • See “SIP Credential” on page 277 for more details.

Country-Specific Configuration

This chapter describes how to set the Mediatrix 4100 with the proper country settings.

Caller ID Information

The caller ID is a generic name for the service provided by telephone utilities that supply information such as the telephone number or the name of the calling party to the called subscriber at the start of a call. In call waiting, the caller ID service supplies information about a second incoming caller to a subscriber already busy with a phone call. However, note that caller ID on call waiting is not supported by all caller ID-capable telephone displays.

In typical caller ID systems, the coded calling number information is sent from the central exchange to the called telephone. This information can be shown on a display of the subscriber telephone set. In this case, the caller ID information is usually displayed before the subscriber decides to answer the incoming call. If the line is connected to a computer, caller information can be used to search in databases and additional services can be offered.

The following basic caller ID features are supported:

- ▶ Date and Time
- ▶ Calling Line Identity
- ▶ Reason for Absence of Calling Line Identity
- ▶ Calling Party Name
- ▶ Reason for Absence of Calling Party Name
- ▶ Visual Indicator (MWI)

Caller ID Generation

There are two methods used for sending caller ID information depending on the application and country-specific requirements:

- ▶ caller ID generation using DTMF signalling
- ▶ caller ID generation using Frequency Shift Keying (FSK)

Both methods can be used on different lines at the same time.

DTMF Signalling

The data transmission using DTMF signalling is performed during or before ringing depending on the country settings or line configuration. The Mediatrix 4100 provides the calling line identity according to the following standards:

- ▶ Europe: ETSI 300 659-1 January 2001 (Annex B) : Access and Terminals (AT); Analogue access to the Public Switched Telephone Network (PSTN); Subscriber line protocol over the local loop for display (and related) services; Part 1: On-hook data transmission.
- ▶ Brazil: STD 220-250-713 Issue 01. November 1993: General specification "identification of the calling party for SPC with DTMF".



Note: For units in Brazil, set the *analogScnGwInterDigitDialDelay* and *analogScnGwDtmfDuration* value to 70 ms (in the *analogScnGwMIB*). This will ensure that the caller ID displays properly.

- ▶ Denmark: TDK-TS 900 301-1 January 2003: Public Switched Telephone Network (PSTN) Calling Line Identification presentation (CLIP) supplementary service Specification of the NTP.

FSK Generation

Different countries use different standards to send caller ID information. The Mediatrix 4100 is compatible with the following widely used standards:

- ▶ Bellcore GR-30-CORE
- ▶ British Telecom (BT) SIN227, SIN242
- ▶ UK Cable Communications Association (CCA) specification TW/P&E/312
- ▶ ETSI 300 659-1



Note: The compatibility of the Mediatrix 4100 is not limited to the above caller ID standards.

Continuous phase binary FSK modulation is used for coding which is compatible with:

- ▶ BELL 202
- ▶ ITU-T V.23, the most common standard

ADSI

ADSI (Analog Display Service Interface) is a telecommunications protocol standard that enables alternate voice and data capability over the existing analog telephone network. It is an extension to basic caller ID. To use ADSI, you would need an ADSI capable device.

ADSI can display the basic caller ID parameters and the following additional parameters:

- ▶ Call Type
- ▶ First Called Line Identity
- ▶ Number of Messages (MWI)
- ▶ Type of Forwarded Call
- ▶ Type of Calling User
- ▶ Redirecting Number
- ▶ Charge
- ▶ Duration of the Call
- ▶ Network Provider Identity



Note: Currently, very few ADSI-capable devices support these additional information.

Setting the Location (Country)

It is very important to set variables according to the country in which the Mediatrix 4100 is used because a number of parameter values are set according to this choice. These parameters are:

- ▶ Tones
- ▶ Rings
- ▶ Impedances
- ▶ Line Attenuations

See ["Appendix D - Country-Specific Parameters" on page 385](#) for more information on these country-specific settings.



In the *Unit Manager Network Administration Manual*, refer to chapter *Ports Parameters*, section *Port Configuration Window*.

You can also set these parameters via the web interface, as described in ["Country Selection" on page 104](#).

► **To set a country location:**

1. In the *telephonyMIB*, locate the *telephonyCountrySelection* variable.
This variable indicates the current country used by the Mediatrix 4100. It can also be used to select a caller ID standard in countries that support more than one caller ID standard.
2. Set the variable with one of the following parameters:

North America 1	Australia 2	New Zealand
North America 2	Australia 3	UAE 2 (United Arab Emirates)
Austria 1	Japan	Czech Republic
Austria 2	Israel	Chile1
France	Thailand	Chile2
Germany 1	Indonesia	South Africa
Germany 2	China	uk-bellcore
Germany 3	Hong Kong	uk-cca
UK	Malaysia	uk-etsi-fsk
Italy	Russia	france-etsi-fsk
Spain	Netherlands	france-etsi-dtmf
Switzerland	Brazil	austria-etsi-fsk
Sweden	Mexico	austria2-etsi-fsk
Australia 1	Denmark	
3. Restart the Mediatrix 4100 so that the changes may take effect.

Caller ID Selection

In countries that support more than one caller ID standard, this standard can be selected with the *telephonyCountrySelection* variable. Be careful to properly select the option corresponding to your caller ID.

Table 111: Caller ID Mappings

Country	Caller ID	<i>telephonyCountrySelection</i> variable Mapping
UK	British Telecom	uk
	Bellcore	uk-bellcore
	CCA	uk-cca
	ETSI-FSK	uk-etsi-fsk
France	Bellcore	france
	ETSI-FSK	france-etsi-fsk
	ETSI-DTMF	france-etsi-dtmf
Austria 1	Bellcore	austria1
	ETSI-FSK	austria-etsi-fsk
Austria 2	Bellcore	austria2
	ETSI-FSK	austria2-etsi-fsk

See ["Caller ID Information" on page 177](#) for more details.

Custom Tone Configuration

You can override the pattern for a specific tone defined for the selected country (see [“Appendix D - Country-Specific Parameters” on page 385](#) for more details). You can define new patterns for the following tones:

- | | |
|-------------------|----------------------------------|
| ▶ Busy | ▶ Preemption |
| ▶ Confirmation | ▶ Reorder |
| ▶ Congestion | ▶ Ringback |
| ▶ Dial | ▶ Receiver Off Hook (ROH) |
| ▶ Intercept | ▶ Special Information Tone (SIT) |
| ▶ Message Waiting | ▶ Stutter |

Pattern Definition

The general format of the pattern string is defined in the following ABNF:

```
tone-pattern = [ frequencies-section [ "," loop-counter-section ] "," states-section ]
```

This general pattern uses the following three main categories

```
frequencies-section = "f1=" frequency-description
                    [ ",f2=" frequency-description
                    [ ",f3=" frequency-description
                    [ ",f4=" frequency-description ] ] ]

loop-counter-section = "l=" loop-counter

states-section = ",s1=" state-description
               [ ",s2=" state-description
               [ ",s3=" state-description
               [ ",s4=" state-description
               [ ",s5=" state-description
               [ ",s6=" state-description
               [ ",s7=" state-description
               [ ",s8=" state-description ] ] ] ] ] ]
```

Finally, the three main categories use the following parameters and tags:

```
frequency-description = frequency ":" power
frequency = 2*4DIGIT
power = ( DIGIT / "-" 1*2DIGIT )
loop-counter = 1*2DIGIT
state-description = ( on-state-description / off-state-description )
on-state-description = "on" frequency-selection [ time ] [ loop-indicator ] [ next-state ]
off-state-description = "off" [ time ] [ loop-indicator ] [ next-state ]
frequency-selection = [ ":f1" ] [ ":f2" ] [ ":f3" ] [ ":f4" ]
time = ":" 2*5DIGIT
loop-indicator = "l"
next-state = ":" ( "s1" / "s2" / "s3" / "s4" / "s5" / "s6" / "s7" )
```

The following table describes the various tags used in the syntax.

Table 112: Pattern Definition Syntax

Tag	Description
tone-pattern	String describing the pattern to use for the tone. An empty string means no tone.
frequencies-section	Description of the frequencies used by the tones used in states-section. You can define up to four frequencies (f1 to f4). You must enter at least one frequency if the tone-pattern is not empty. The frequencies to use are defined in the state-description.
frequency-description	Description of the frequency. This is described as "frequency:power".
frequency	Frequency value in Hz. The range is from 10 Hz to 4000 Hz.
power	Power level of the frequency in dBm. The range is from -99 dBm to 3 dBm.
loop-counter-section	Loop counters definition. The loop counter is used in state-description.
loop-counter	Value of the loop counter. The range is from 2 to 128.
states-section	Description of the tone state. You can define up to eight states (s1 to s8). You must enter at least one state if the tone-pattern is not empty.
state-description	Description of the tone state.
on-state-description	Description of a state playing a tone.
off-state-description	Description of a state not playing a tone.
frequency-selection	Frequency to play in the state. You can use from one to four frequencies. The frequency must be defined in frequencies-section.
time	The number of times, in ms, to perform the action of the state. The range is from 10 ms to 56000 ms. The tone stays indefinitely in the state if no time is specified.
loop-indicator	Used to stop looping between states after a number of loops defined in loop-counter-section. When the number of loops is reached, the next state is s(n+1) for the state s(n) instead of the state defined in next-state.
next-state	The next tone state to use when the time has elapsed. This value is not present if the time is not present.

Customizing the Tones

The *Custom Tone* section allows you to define new patterns as per the pattern syntax.

► To customize one or more tones:

1. In the *telephonyMIB*, locate the *countryCustomizationToneTable* table.
2. Define whether or not you want to override the default tone configuration for a specific tone by setting the *countryCustomizationToneOverride* variable.
3. Enter the override pattern in the corresponding *countryCustomizationToneTone* variable.
You must follow the syntax as described in ["Pattern Definition" on page 180](#).
See ["Custom Tone Example" on page 182](#) for a detailed example on how to create a proper pattern.

The following table gives some examples of custom tones. Note that the quotation marks are not part of the syntax and must not be included when entering the tone pattern.

Table 113: Pattern Examples

Example	Pattern
No tone	""
North America dial tone (continuous tone at 350 Hz and 440 Hz with a -17 power level)	"f1=350:-17,f2=440:-17,s1=on:f1:f2"
North America Recall dial tone (three quick tones followed by a continuous tone)	"f1=350:-17,f2=440:-17,l=3,s1=on:f1:f2:100:s2,s2=off:100:l:s1,s3=on:f1:f2"
Australia ring back tone (tone on 400 ms, off 200 ms, on 400 ms, and off 2000 ms and replay)	"f1=425:-17,f2=400:-5,f3=450:-5,s1=on:f1:f2:f3:400:s2,s2=off:200:s3,s3=on:f1:f2:f3:400:s4,s4=off:2000:s1"

Custom Tone Example

This section describes how to create the pattern for the North America recall dial tone (also called stutter dial tone), which is three quick tones followed by a continuous tone.

```
"f1=350:-17,f2=440:-17,l=3,s1=on:f1:f2:100:s2,s2=off:100:l:s1,s3=on:f1:f2"
```

► To create the pattern:

- Let's start with the general format of the pattern string:

```
Pattern = [ frequencies-section [ "," loop-counter-section ] "," states-section ]
```

- Set the frequencies-section category, which is defined as follows:

```
frequencies-section = "f1=" frequency-description
                    [ ",f2=" frequency-description
                    [ ",f3=" frequency-description
                    [ ",f4=" frequency-description ] ] ]
```

- The frequency-description parameter is described as follows:

```
frequency:power
```

- In the North America stutter dial tone, two frequencies are used, 350 Hz and 440 Hz. Their power level is -17 dBm. You can thus complete the frequencies-section category as follows:

```
frequencies-section = "f1=" 350:-17 ",f2=" 440:-17 ]
```

- The general format of the pattern string now looks as follows:

```
Pattern = [ f1=350:-17,f2=440:-17 [ "," loop-counter-section ] "," states-section ]
```

- Set the loop-counter-section category, which is defined as:

```
loop-counter-section = "l=" loop-counter
```

It defines the number of times to repeat the pattern.

- a. The loop-counter part is defined as follows:

```
loop-counter = 1*2DIGIT
```

- b. In the North America stutter dial tone, the pattern is repeated three times, thus:

```
loop-counter = 3
```

- c. The loop-counter-section category now looks as follows:

```
loop-counter-section = "l=" 3
```

- d. The general format of the pattern string now looks as follows:

```
Pattern = [ f1=350:-17,f2=440:-17 [ "," "l=" 3 ] "," states-section ]
```

4. Set the states-section category, which is defined as:

```
states-section = ",s1=" state-description
                [ ",s2=" state-description
                [ ",s3=" state-description
                [ ",s4=" state-description
                [ ",s5=" state-description
                [ ",s6=" state-description
                [ ",s7=" state-description
                [ ",s8=" state-description ] ] ] ] ] ]
```

- a. state-description is defined as:

```
state-description = ( on-state-description / off-state-description )
```

- b. There are three states in the North America stutter dial tone: 0.1 on, 0.1 off, and continuous. The pattern that must be described is thus:

```
states-section = ",s1=" state-description
                [ ",s2=" state-description
                [ ",s3=" state-description ] ]
```

5. Let's define the first state. Since the first state describes an on tone, off-state-description is not required for this state.

- a. You now have to complete the on-state-description parameter for the first state, which is defined as:

```
on-state-description = "on" frequency-selection [ time] [ loop-indicator ] [
next-state ]
```

- b. frequency-selection is defined as the frequencies to play and has the following syntax:

```
frequency-selection = [ ":f1" ] [ ":f2" ] [":f3" ] [ ":f4" ]
```

You can use from one to four frequencies. The North America stutter dial tone has two frequencies, thus:

```
frequency-selection = [ ":f1" ] [ ":f2" ]
```

- c. The on-state-description parameter is now:

```
on-state-description = "on" [ ":f1" ] [ ":f2" ] ] [ time] [ loop-indicator ]
[ next-state ]
```

- d. The time parameter is defined as:

```
":" 2*5DIGIT
```

It is the number of milliseconds to perform the action of the state. The on state is 100 ms, thus,

```
on-state-description = "on" [ ":f1" ] [ ":f2" ] [ :100 ] [ loop-indicator ] [
next-state ]
```

- e. The loop-indicator parameter is not used in this state. The on-state-description parameter is now:

```
on-state-description = "on" [ ":f1" ] [ ":f2" ] [ :100 ] [ next-state ]
```

- f. The next-state parameter is defined as:

```
next-state = ":" ( "s1" / "s2" / "s3" / "s4" / "s5" / "s6" / "s7" )
```

It is the next tone state to use when the time has elapsed. In this case, the next-state parameter is the off state, which is designated as s2.

The on-state-description parameter is now:

```
on-state-description = "on" [ ":f1" ] [ ":f2" ] [ :100 ] [ :s2 ]
```

- g. You can now complete the first state-description parameter:

```
states-section = ",s1=" ( on-state-description)
                  [ ",s2=" state-description
                  [ ",s3=" state-description ] ]
```

becomes:

```
states-section = ",s1=" ( "on" [ ":f1" ] [ ":f2" ] [ :100 ] [ :s2 ] )
                  [ ",s2=" state-description
                  [ ",s3=" state-description ] ]
```

6. Let's define the second state. Since the first state describes an off tone, on-state-description is not required for this state.

- a. You now have to complete the off-state-description parameter for the first state.

The off-state-description parameter is defined as:

```
off-state-description = "off" [ time ] [ loop-indicator ] [ next-state]
```

- b. The time parameter is defined as:

```
":" 2*5DIGIT
```

It is the number of milliseconds to perform the action of the state. The off state is 100 ms, thus,

```
off-state-description = "off" [ :100 ] [ loop-indicator ] [ next-state]
```

- c. The loop-indicator parameter is defined as:

```
loop-indicator = ":l"
```

It is used to stop looping between states. It indicates that the loop stops after three times. Once the loop is completed, the pattern goes to the next state (which is state 3).

The off-state-description parameter is now:

```
off-state-description = "off" [ :100 ] [ :1 ] [ next-state ]
```

- d. The next-state parameter is defined as:

```
next-state = ":" ( "s1" / "s2" / "s3" / "s4" / "s5" / "s6" / "s7" )
```

It is the next tone state to use when the time has elapsed. In this case, the next-state parameter is the on state, which is designated as s1.

The off-state-description parameter is now:

```
off-state-description = "off" [ :100 ] [ : 1 ] [ :s1 ]
```

- e. You can now complete the second state-description parameter:

```
states-section = ",s1=" ( "on" [ ":f1" ] [ ":f2" ] [ :100 ] [ :s2 ] )
                  [ ",s2=" state-description
                    [ ",s3=" state-description ] ]
```

becomes:

```
states-section = ",s1=" ( "on" [ ":f1" ] [ ":f2" ] [ 100 ] [ s2 ] )
                  [ ",s2=" "off" [ :100 ] [ :1 ] [ :s1 ]
                    [ ",s3=" state-description ] ]
```

7. Let's define the third and last state. Since the third state describes an on tone, off-state-description is not required for this state.

- a. You now have to complete the on-state-description parameter for the first state.

The on-state-description parameter is defined as:

```
on-state-description = "on" frequency-selection [ time ] [ loop-indicator ] [
next-state ]
```

- b. frequency-selection is defined as the frequencies to play and has the following syntax:

```
frequency-selection = [ ":f1" ] [ ":f2" ] [ ":f3" ] [ ":f4" ]
```

You can use from one to four frequencies. The North America stutter dial tone has two frequencies, thus:

```
frequency-selection = [ ":f1" ] [ ":f2" ]
```

- c. The on-state-description parameter is now:

```
on-state-description = "on" [ ":f1" ] [ ":f2" ] [ time ] [ loop-indicator ] [
next-state ]
```

- d. The time parameter is the number of milliseconds to perform the action of the state. Since the third state is a continuous tone, the time parameter is not required, thus,

```
on-state-description = "on" [ ":f1" ] [ ":f2" ] [ loop-indicator ] [ next-
state ]
```

- e. The loop-indicator parameter is used to stop looping between states. Since the third state is a continuous tone and does not use loops, this parameter is not required.

The on-state-description parameter is now:

```
on-state-description = "on" [ ":f1" ] [ ":f2" ] [ next-state ]
```

- f. The next-state parameter is the next tone state to use when the time has elapsed. This value is not present if the time parameter is not present. You have already discarded the time parameter, so the next-state parameter is not required.

The on-state-description parameter is now:

```
on-state-description = "on" [ ":f1" ] [ ":f2" ]
```

- g. You can now complete the third state-description parameter and the states-section parameter:

```
states-section = ",s1=" ( "on" [ ":f1" ] [ ":f2" ] [ :100 ] [ :s2 ] )
                  [ ",s2=" "off" [ :100 ] [ :1 ] [ :s1 ]
                  [ ",s3=" state-description ] ]
```

becomes:

```
states-section = ",s1=" ( "on" [ ":f1" ] [ ":f2" ] [ :100 ] [ :s2 ] )
                  [ ",s2=" "off" [ :100 ] [ :1 ] [ :s1 ]
                  [ ",s3=" "on" [ ":f1" ] [ ":f2" ] ] ]
```

8. Now that you have the three main categories completed, you can finish the pattern:

```
Pattern = [ f1=350:-17,f2=440:-17 [ ", " "l=" 3 ] ", " ",s1=" ( "on" [ ":f1" ]
[ ":f2" ] [ :100 ] [ :s2 ] )
            [ ",s2=" "off" [ :100 ] [ :1 ] [ :s1 ]
            [ ",s3=" "on" [ ":f1" ] [ ":f2" ] ] ] ]
```

If you remove all the brackets and quotation marks, which are not to be included, the pattern is:

```
Pattern = f1=350:-17,f2=440:-
17,l=3,s1=on:f1:f2:100:s2,s2=off:100:1:s1,s3=on:f1:f2
```

Ther pattern could also be defined as follows:

```
Pattern = f1=350:-17,f2=440:-
17,s1=on:f1:f2:100:s2,s2=off:100:s3,s3=on:f1:f2:100:s4,s4=off:100:s5,s5=on:f
1:f2:100:s6,s6=off:100:s7,s7=on:f1:f2
```


Configuration File Download

The configuration file download feature allows to update the Mediatrix 4100 configuration by transferring a configuration file via TFTP, HTTPS, or HTTP. The configuration file can either be transferred from the management server or from the configuration file download server. The main difference is the session initiator, which is respectively the management server and the Mediatrix 4100. The advantage of having the Mediatrix 4100 as the session initiator is to allow NAT traversal.

You can also set these parameters via the web interface, as described in [“Configuration File Download” on page 40](#).

Configuration File Download Server

The service allows to download a unique file for each Mediatrix 4100, and/or a file shared among many units. These configuration files may be encrypted or not.

You have the choice to perform the configuration file download by using the TFTP protocol, the HTTPS protocol or the HTTP protocol. You can also configure the Mediatrix 4100 to automatically update its configuration.

To download a configuration file, you may need to setup the following applications on your computer:

- ▶ TFTP server with proper root path
- ▶ SNTP server properly configured
- ▶ HTTP server with proper root path
- ▶ HTTPS server with proper root path
- ▶ Configuration source
- ▶ Configuration file name and location

Configuring the TFTP Server

If you are to perform a configuration file download by using the TFTP protocol, you must install a TFTP (Trivial File Transfer Protocol) server running on the PC designated as the TFTP server host. It is assumed that you know how to set the TFTP root path. If not, refer to your TFTP server's documentation.

Configuring the SNTP Server

If you are to use the automatic configuration file update feature (see [“Automatic Configuration Update” on page 196](#) for more details) or the HTTPS protocol, you need to have a time server SNTP that is accessible and properly configured. It is assumed that you know how to configure your SNTP server. If not, refer to your SNTP server's documentation. You can also refer to [“Chapter 20 - SNTP Settings” on page 289](#) for more details on how to configure the Mediatrix 4100 for a SNTP server.

Configuring the HTTP Server

If you are to perform a configuration file download by using the HTTP protocol, you must install a HTTP server running on the PC designated as the server host. It is assumed that you know how to set the root path. If not, refer to your HTTP server's documentation.

Configuring the HTTPS Server

Standards Supported

- RFC 2246 – The TLS Protocol Version 1.0
- RFC 2818 – HTTP Over TLS
- RFC 2459 – X.509 Digital Certificates
- RFC 3280 – Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile

If you are to perform a configuration file download that requires authentication or privacy by using the HTTP over the Transport Layer Security (TLS) protocol (HTTPS), you must install a HTTPS server running on the PC designated as the server host. It is assumed that you know how to set the root path and set the SSL/TLS security configuration. If not, refer to your HTTPS server's documentation.



Note: The web interface does not support the HTTPS protocol.



Caution: You must have a time server SNTP that is accessible and properly configured, or the automatic configuration update feature may not work properly. It is assumed that you know how to configure your SNTP server. If not, refer to your SNTP server's documentation. You can also refer to [“Chapter 20 - SNTP Settings” on page 289](#) for more details on how to configure the Mediatrix 4100 for a SNTP server.

The Mediatrix 4100 supports the following SSL/TLS parameters:

Table 114: Secure Parameters Supported

Supported Parameter	Description
Key Exchange Mechanism	• RSA • Diffie-Hellman
Ciphers	• AES (128 and 256 bits) • 3DES (168 bits)
Message Digests	• SHA-1



Note: Media5 recommends to use cipher suites based on the RSA key exchange mechanism, because the Diffie-Hellman key exchange mechanism introduces a noticeable delay in the HTTPS session establishment.

Certificates

The Mediatrix 4100 contains embedded security certificates formatted as per ITU x.509 and RFC 3280. The certificates are factory-installed. The name of X.509 certificates currently installed in the Mediatrix 4100 are listed in the *securityCertificatesMIB* under the *certificateName* variable (under the *certificateTable* group). You must have at least one matching certificate on your HTTPS server.

You can also find the following information:

Table 115: Certificates

Parameter	Description
certificateSubjectCommonName	The certificate's subject name. If the certificate contains a subject field, display the common name. Otherwise display the first non-empty subject alternative name from the following list: • Uniform Resource Locator • DNS name • IP Address • RFC 822 name See RFC 3280 sections 4.1.2.6 and 4.2.1.7 for details.
certificateExpirationDate	The certificate's expiration date. Display the date at which the certificate expires. The format is MM/DD/YYYY in universal time. See RFC 3280 section 4.1.2.5 for details.

When contacting a HTTPS server, the Mediatrix 4100 establishes a TLS connection by (among others):

- ▶ negotiating cipher suites
- ▶ checking the server certificates validity (dates)

The Mediatrix 4100 then checks the server's identity by validating the host name used to contact it against the information found in the server's certificate, as described in RFC 2818, section 3.1.

If any of the above does not succeed, the Mediatrix 4100 refuses the secure connection. To help detect such errors, you can increase the syslog messages level.

Configuration File Server Settings

The Mediatrix 4100 must know the IP address and port number of its configuration file server. This server contains the configuration file the Mediatrix 4100 will download. You can assign these information to the Mediatrix 4100 through a DHCP server or manually enter them yourself in static variables.

DHCP Configuration

Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See ["Chapter 8 - IP Address and Network Configuration" on page 143](#) for more details.

▶ To use DHCP-assigned information:

1. In the *ipAddressConfig* folder, locate the *configFileFetchingSelectConfigSource* variable (under the *ipAddressConfigFileFetching* group).
This variable defines whether the Mediatrix 4100 must ask for its configuration file server settings through a DHCP server or not.
2. Set the *configFileFetchingConfigSource* variable to **dhcp**.
You can query the configuration file server's IP address and port number assigned by the DHCP server in the following read-only variables (in the *ipAddressStatus* folder):
 - *configFileFetchingHost*
 - *configFileFetchingPort*

3. Set how you want to define the configuration server information in the DHCP server:

Table 116: Configuration File Server DHCP Information

To use a...	Set...
vendor specific code	The <i>configFileFetchingDhcpSiteSpecificCode</i> variable to 0 . Set the configuration file server IP address in the DHCP server inside the vendor specific sub-option 201 (hexadecimal 0xC9).
site specific code	The <i>configFileFetchingDhcpSiteSpecificCode</i> variable to any value between 128 and 254. Set the configuration file server IP address in the DHCP server inside the site specific option you have chosen (it must match the value of the <i>configFileFetchingDhcpSiteSpecificCode</i> variable in the unit's configuration).

See ["Vendor and Site Specific DHCP Options" on page 156](#) for more details.

Static Configuration

Use the static configuration if you are not using a DHCP server or if you want to bypass it.

► To use static information:

1. In the *ipAddressConfig* folder, locate the *configFileFetchingSelectConfigSource* variable. This variable defines whether the Mediatrix 4100 must ask for its configuration file server settings through a DHCP server or not.
2. Set the *configFileFetchingSelectConfigSource* variable to **static**.
3. Set the following variables:

Table 117: Configuration File Server Static Information

Variable	Description
<i>configFileFetchingStaticHost</i>	Static configuration file server IP address or domain name to use when downloading a configuration file. This is the current address of the PC that hosts the configuration files. Default Value: 192.168.0.10
<i>configFileFetchingStaticPort</i>	Static configuration file server IP port number to use when downloading a configuration file. Default Value: 69

The default port value complies to RFC 1340 on the well-known ports (assigned numbers). This value applies to a TFTP server. It may be different for other servers. If you are using an HTTP/HTTPS server to perform the configuration file download, you must change the port value to 80.

Setting up the Configuration File Download

When performing a configuration file download, you can download two different files:

- A generic configuration file that should be used to update a large number of units with the same configuration.
- A specific configuration file that contains the configuration for a single unit, for instance the telephone numbers of its lines.

When both the generic and specific configuration files are downloaded, settings from the specific configuration file always override the settings from the generic configuration file. These files must be located in the same directory.

► To setup the configuration file download:

1. In the *configFileFetchingMIB*, set the *configFileFetchingFileLocation* variable with the path, on the remote server, of the directory where the configuration files are located.

The path is case sensitive hence it must be entered properly.

The path is relative to the root path of the transfer server (*configFileFetchingHost*). Use the "/" character when defining the path to indicate sub-directories.

Let's consider the following example:

- The directory that contains the configuration file is called: **Config_File**.
- This directory is under **C:/Root/Download**.

Table 118: Path Configurations Example

Root Path	Corresponding Path Name
c:/root/download	Config_File
c:/	root/download/Config_File
c:/root	download/Config_File

The following are some tips to help your download process:

- Use the "/" character when defining the path to indicate sub-directories. For instance, *root/download*.
- If you are using the TFTP protocol to download the software, note that some TFTP servers on Windows do not recognize the "/" character and produce an error. In this case, use the "\" character.
- Use basic directory names, without spaces or special characters such as "~", "@", etc., which may cause problems.
- Cut and paste the path and/or name of the directory that contains the extracted files into the configuration file path of the Mediatrix 4100 (you may have to convert "\" into "/" to eliminate typographical errors).

Note that you can define the **C:/Root/Download** part as you want. The file names may also differ from the example shown above.

2. Set the *configFileFetchingFileName* variable with the name of the generic configuration file to download.



Caution: The generic configuration file must be in XML format, no matter what its file extension.

The file name is case sensitive hence it must be entered properly.

This file should be used to update a large number of units with the same configuration.

If you leave the variable empty, the Mediatrix 4100 does not download the generic configuration file.

3. Set the *configFileFetchingSpecificFileName* variable with the name of the specific configuration file to download.



Caution: The specific configuration file must be in XML format, no matter what its file extension.

The file name is case sensitive hence it must be entered properly.

This file should be used to update the configuration of a single unit.

This variable may contain macros that are substituted by actual values when downloading the configuration file. Supported macros are:

- %mac%: the MAC address of the unit
- %product%: the product name of the unit

- %%: the character “%”

For instance:

- The “%mac%.xml” value for a Mediatrix 4100 with MAC address “0090F12345AB” will be “0090F12345AB.xml”.
- The value “Hello%%Hi” will result in “Hello%Hi”.
- The value “%%mac%%mac%.xml” will result in “%0090F12345AB%mac%.xml”.

From left to right: the first macro encountered is first substituted, the second macro encountered is then substituted, etc.

When the character “%” is not part of a macro, it is not replaced. The following are examples:

- The value “%mac.xml” stays “%mac.xml”
- The value “Hello%Hi” stays “Hello%Hi”
- The value “%moc%.xml” stays “%moc%.xml”

If the variable is empty (after macro substitution), the Mediatrix 4100 does not download the specific configuration file.

Configuration Update Status

If valid configuration files are successfully downloaded, then the Mediatrix 4100 automatically restarts to apply all the new settings. If the Mediatrix 4100 does not restart, this could mean the download failed or that the configuration in the file is the same as the configuration in the unit.

You can validate the status of the configuration update in various ways.

MIB Variable

You can query the status of the last configuration file download in the *sysAdminDownloadConfigFileStatus* variable:

- ▶ idle: No configuration file download has been performed yet.
- ▶ fail: The last configuration file download failed.
- ▶ success: The last configuration file download succeeded.
- ▶ inProgress: A configuration file download is in progress.
- ▶ listening: The unit is listening and waiting for a configuration file to be sent by the management server.

Syslog Messages

A lot of information is transmitted as system log (syslog) messages. The following are some of the syslog messages sent by the unit:

Table 119: Configuration File Download Syslog Messages

Level	Message	Event
Informational	The specific configuration update succeeded.	The configuration update with the specific configuration file has been successful.
Error	The specific configuration update failed.	The configuration update with the specific configuration file experienced an error and has not been completed.
Informational	The configuration file “XXX” was successfully fetched.	A configuration file was successfully fetched.
Informational	The unit configuration is not updated. The parameter values defined in the fetched configuration files are identical to the actual unit configuration.	The parameter values defined in the fetched configuration files are identical to the actual unit configuration.

Table 119: Configuration File Download Syslog Messages (Continued)

Level	Message	Event
Informational	The generic file \"%s\" parameter values are not applied. They are either identical to the unit configuration or overwritten by the specific file.	The generic configuration file parameter values are either identical to the unit configuration or overwritten by the specific configuration file.
Warning	None of the parameter values defined in the configuration file \"%s\" was successfully applied.	No parameter value from a fetched configuration file was successfully applied (e.g., because of bad OIDs).
Informational	Parameter values defined in the configuration file \"%s\" were successfully applied.	A fetched configuration file was successfully applied.
Informational	The unit is restarting to complete the configuration update.	All necessary fetched configuration files were successfully applied.

Configuration Files Encryption

You can secure the exchange of configuration files between the server and the Mediatrix 4100. A privacy key allows the unit to decrypt a previously encrypted configuration file. This applies to files downloaded via TFTP, HTTPS, or HTTP.

To encrypt a configuration file (generic or specific), you must use the MxCryptFile application. MxCryptFile is a command line tool that encrypts files before sending them to the Mediatrix 4100 unit. Contact your sales representative for more details.

Configuration File Decryption on the Mediatrix 4100

The following describes how to decrypt a previously encrypted generic or specific configuration file. You must have one key for the generic configuration file and another key for the specific configuration file.

► To decrypt a configuration file:

1. In the *configFileFetchingMIB*, set the proper decryption variable with the secret key used to decrypt the configuration file.

Table 120: Decryption Variables

Configuration File	Variable
Generic	configFilePrivacyGenericSecret
Specific	configFilePrivacySpecificSecret

The key is encoded in hexadecimal notation. You can thus use characters in the range 0-9, A-F. All other characters are not supported.

Each character encodes 4 bits of the key. For instance, a 32-bit key requires 8 characters.

- If you enter too many bits, the key is truncated to the first 448 bits.
- If you do not enter enough bits, the key is padded with zeros.

For instance, a 32-bit key could look like the following: A36CB299.

This key must match the key used for the encryption of the relevant configuration file.

If the variable is empty, the configuration file is not decrypted.

2. Set the *configFilePrivacyEnable* variable to **enable**.

The Mediatrix 4100 will be able to decrypt the next encrypted generic or specific configuration file. If this variable is set to **disable**, the configuration file is not decrypted by the unit and the configuration update fails.

Configuration Download via TFTP

The following steps explain how to download configuration files by using the TFTP protocol.



Note: The configuration download via TFTP can only traverse NATs of types “Full Cone” or “Restricted Cone”. If the NAT you are using is of type “Port Restricted Cone” or “Symmetric”, the file transfer will not work.

► To download configuration files via TFTP:

1. Set the configuration file server host and port as defined in [“Configuration File Server Settings” on page 189](#).
2. Place the configuration files to download on the computer hosting the TFTP server. These files must be in a directory under the TFTP root path.
3. If not already done, set the configuration file path as described in [“Setting up the Configuration File Download” on page 190](#).
4. In the *configFileFetchingMIB*, set the *configFileTransferProtocol* variable to **tftp**.
5. In the *groupAdminMIB*, set the *groupSetAdmin* variable to **ForceLock**.
All activities in progress on the Medatrix 4100 are terminated immediately and the unit enters the maintenance mode (the value of the *groupAdminState* variable is “locked”). The configuration file download may take place.
6. In the *sysAdminMIB*, initiate the configuration file download via TFTP by setting the *sysConfigCommand* variable to **updateConfiguration**.
The Medatrix 4100 immediately downloads the configuration files. It is the initiator of the TFTP sessions.

NAT Variations

NAT treatment of UDP varies among implementations. The four treatments are:

- Full Cone: All requests from the same internal IP address and port are mapped to the same external IP address and port. Furthermore, any external host can send a packet to the internal host by sending a packet to the mapped external address.
- Restricted Cone: All requests from the same internal IP address and port are mapped to the same external IP address and port. Unlike a full cone NAT, an external host (with IP address X) can send a packet to the internal host only if the internal host had previously sent a packet to IP address X.
- Port Restricted Cone: Similar to a restricted cone NAT, but the restriction includes port numbers. Specifically, an external host can send a packet, with source IP address X and source port P, to the internal host only if the internal host had previously sent a packet to IP address X and port P.
- Symmetric: All requests from the same internal IP address and port, to a specific destination IP address and port, are mapped to the same external IP address and port. If the same host sends a packet with the same source address and port, but to a different destination, a different mapping is used. Furthermore, only the external host that receives a packet can send a UDP packet back to the internal host.

For more details on NAT treatments, refer to RFC 3489.

Configuration Download via HTTP/HTTPS

The following steps explain how to download the configuration files by using the HTTP or HTTPS protocol. If you are using HTTPS, the Mediatrix 4100 must contain the proper certificate. See [“Configuring the HTTPS Server” on page 188](#) for more details.

► To download the configuration files via HTTP or HTTPS:

1. Set the configuration file server host and port as defined in [“Configuration File Server Settings” on page 189](#).



Caution: When downloading via HTTP or HTTPS, the configuration file server's port must be 80. You can query the actual port assigned in the *configFileFetchingPort* read-only variable (in the *ipAddressStatus* folder).

If you are using a DHCP server and it did not provide the proper port, reconfigure it with the proper port or use a static configuration. See [“Configuration File Server Settings” on page 189](#) for more details.

2. Place the configuration files to download on the computer hosting the HTTP or HTTPS server. These files must be in a directory under the root path.
3. If not already done, set the configuration file path as described in [“Setting up the Configuration File Download” on page 190](#).
4. In the *configFileFetchingMIB*, set the *configFileTransferProtocol* variable to **http** or **https**.
Your HTTP or HTTPS server may activate some caching mechanism for the file download. This mechanism caches the initial file download for later processing, thus preventing changes or update of the original file by the user. This can cause strange problems if a user wants to edit a configuration file to modify values and upload it immediately. The result will still return the original file and not the new one.
5. If your HTTP or HTTPS server requires authentication when downloading the configuration file, set the following:
 - The user name in the *configFileTransferUsername* variable.
 - The password in the *configFileTransferPassword* variable.
6. In the *groupAdminMIB*, set the *groupSetAdmin* variable to **ForceLock**.
All activities in progress on the Mediatrix 4100 are terminated immediately and the unit enters the maintenance mode (the value of the *groupAdminState* variable is “locked”). The configuration file download may take place.
7. In the *sysAdminMIB*, initiate the configuration file download via HTTP or HTTPS by setting the *sysConfigCommand* variable to **updateConfiguration**.
The Mediatrix 4100 immediately downloads the configuration files. It is the initiator of the HTTP/HTTPS sessions.

User Agent Header of HTTP Requests

The *User-Agent* header field of an HTTP request contains information about the user agent client originating the request. For instance, the information of the *User-Agent* header could be something like the following:

User-Agent: Softphone Beta1.5

You can customize the information that the Mediatrix 4100 sends when establishing a communication.

► To customize the HTTP User Agent header of HTTP requests:

1. In the *interopMIB*, set the *mxInteropHttpUAHeaderConfig* variable with the proper macro.
The following macros are replaced by their representation:
 - **%version%**: Version of the application.
 - **%mac%**: Unit MAC address (lowercase).
 - **%rev%**: Hardware revision number.
 - **%product%**: Product name.

- %%: A '%' sign.
2. Restart the Mediatrix 4100 so that the changes may take effect.

Automatic Configuration Update

You can configure the Mediatrix 4100 to automatically update its configuration. This update can be done:

- ▶ Every time the Mediatrix 4100 restarts.
- ▶ At a specific time interval you can define.

Automatic Update on Restart

The Mediatrix 4100 may download new configuration files each time it restarts.

▶ **To set the automatic update every time the Mediatrix 4100 restarts:**

1. Set the configuration file server host and port as defined in ["Configuration File Server Settings" on page 189](#).



Caution: When downloading via HTTP or HTTPS, the configuration file server's port must be 80. You can query the actual port assigned in the *imagePrimaryPort* and *imageSecondaryPort* read-only variables (in the *ipAddressStatus* folder).

If you are using a DHCP server and it did not provide the proper port, reconfigure it with the proper port or use a static configuration. See ["Configuration File Server Settings" on page 189](#) for more details.

2. Place the configuration files to download on the computer hosting the HTTP, HTTPS, or TFTP server.
These files must be in a directory under the root path.
3. If not already done, set the configuration file path as described in ["Setting up the Configuration File Download" on page 190](#).
4. In the *configFileFetchingMIB*, set the *configFileTransferProtocol* variable to either **http**, **https**, or **tftp**.
If you are using the HTTPS protocol, the Mediatrix 4100 must contain the proper certificate. See ["Configuring the HTTPS Server" on page 188](#) for more details.
If you are using the HTTP or HTTPS protocol to download the configuration, be aware that your HTTP or HTTPS server may activate some caching mechanism for the file download. This mechanism caches the initial file download for later processing, thus preventing changes or update of the original file by the user. This can cause strange problems if a user wants to edit a configuration file to modify values and upload it immediately. The result will still return the original file and not the new one.
5. If you are using the HTTP or HTTPS protocol to download the configuration and your HTTP or HTTPS server requires authentication, set the following:
 - The user name in the *configFileTransferUsername* variable.
 - The password in the *configFileTransferPassword* variable.
 The Mediatrix 4100 supports basic and digest HTTP authentication, as described in RFC 2617.
6. Set the *configFileAutoUpdateOnRestartEnable* variable to **enable** (in the *configFileAutomaticUpdate* group).
7. In the *sysConfigMIB*, set the *sysConfigDownloadConfigFile* variable to **automaticInitiateFileDownload**.
The automatic configuration update will be performed each time the Mediatrix 4100 restarts.
The unit configuration is only updated if at least one parameter value defined in the downloaded configuration files is different from the actual unit configuration.

Automatic Update at a Specific Time Interval

You can configure the Mediatrix 4100 to download new configuration files at a specific day and/or time.

► To set the automatic update at a specific time interval:

1. Set the configuration file server host and port as defined in ["Configuration File Server Settings" on page 189](#).



Caution: When downloading via HTTP or HTTPS, the configuration file server's port must be 80. You can query the actual port assigned in the *imagePrimaryPort* and *imageSecondaryPort* read-only variables (in the *ipAddressStatus* folder).

If you are using a DHCP server and it did not provide the proper port, reconfigure it with the proper port or use a static configuration. See ["Configuration File Server Settings" on page 189](#) for more details.

2. Place the configuration files to download on the computer hosting the HTTP, HTTPS, or TFTP server.
These files must be in a directory under the root path.
3. If not already done, set the configuration file path as described in ["Setting up the Configuration File Download" on page 190](#).
4. In the *configFileFetchingMIB*, set the *configFileTransferProtocol* variable to either **http**, **https**, or **tftp**.
If you are using HTTPS, the Mediatrix 4100 must contain the proper certificate. See ["Configuring the HTTPS Server" on page 188](#) for more details.
If you are using the HTTP or HTTPS protocol to download the configuration, be aware that your HTTP or HTTPS server may activate some caching mechanism for the file download. This mechanism caches the initial file download for later processing, thus preventing changes or update of the original file by the user. This can cause strange problems if a user wants to edit a configuration file to modify values and upload it immediately. The result will still return the original file and not the new one.
5. If you are using the HTTP or HTTPS protocol to download the configuration and your HTTP or HTTPS server requires authentication, set the following:
 - The user name in the *configFileTransferUsername* variable.
 - The password in the *configFileTransferPassword* variable.

The Mediatrix 4100 supports basic and digest HTTP authentication, as described in RFC 2617.
6. Define the time base for automatic configuration updates in the *configFileAutoUpdateTimeUnit* variable (in the *configFileAutomaticUpdate* group).
You have the following choices:

Table 121: Time Unit Parameters

Parameter	Description
minutes	Updates the unit's configuration every x minutes. You can specify the x value in the variable <i>configFileAutoUpdatePeriod</i> (see Step 7).
hours	Updates the unit's configuration every x hours. You can specify the x value in the variable <i>configFileAutoUpdatePeriod</i> (see Step 7).
days	Updates the unit's configuration every x days. You can specify the x value in the variable <i>configFileAutoUpdatePeriod</i> (see Step 7). You can also define the time of day when to perform the update in the <i>configFileAutoUpdateTimeRange</i> variable (see Step 8).

7. Set the waiting period between each configuration update in the *configFileAutoUpdatePeriod* variable.
The time unit for the period is specified by the *configFileAutoUpdateTimeUnit* variable (see Step 6). Available values are from 1 to 48.

8. If you have selected **days** in Step 6, set the time of the day when to initiate a configuration update in the `configFileAutoUpdateTimeRange` variable.

The time of the day is based on the `sntpTimeZoneString` variable setting (see [“Chapter 20 - SNTP Settings” on page 289](#) for more details).

You must have a time server SNTP that is accessible and properly configured, or the automatic configuration update feature may not work properly. It is assumed that you know how to configure your SNTP server. If not, refer to your SNTP server's documentation. You can also refer to [“Chapter 20 - SNTP Settings” on page 289](#) for more details on how to configure the Mediatrix 4100 for a SNTP server.

If a time range is specified, the unit will download the configuration files at a random time within the interval specified.

The format should be one of the following:

```
hh[:mm[:ss]]
hh[:mm[:ss]] - hh[:mm[:ss]]
```

Where:

hh: Hours.
mm: Minutes.
ss: Seconds.

The configuration files are downloaded at the first occurrence of this value and thereafter with a period defined by the `configFileAutoUpdatePeriod` variable. Let's say for instance the automatic unit configuration update is set with the time of day at 14h00 and the update period at every 2 days.

- If the automatic update is enabled before 14h00, the first update will take place the same day at 14h00, then the second update two days later at the same hour, and so on.
- If the time range is set to '14:00 - 15:00' and the automatic unit configuration update is enabled within those hours, the first update will take place the following day. This means that a range of '00:00:00 - 23:59:59' will always take place the next day.

9. Set the `configFileAutoUpdatePeriodicEnable` variable to **enable**.
10. In the `sysConfigMIB`, set the `sysConfigDownloadConfigFile` variable to **automaticInitiateFileDownload**.

The unit configuration is only updated if at least one parameter value defined in the downloaded configuration files is different from the actual unit configuration.

If one of the telephones/faxes is off-hook, the Mediatrix 4100 will perform the update 5 minutes after both ports are detected on-hook.

Error Handling

The following configuration file fetching service error sources are divided in three types depending on the transfer protocol: common errors (Table 35), TFTP errors (Table 36) and HTTP/HTTPS errors (Table 37). The error cause and the unit behaviour are also described.

Table 122: Configuration File Fetching Error Handling

Error Type	Cause	Behaviour
Common Error Handling		
Invalid file format	The file format is not valid.	Send a syslog warning message including the file location/name with the transfer server address: The fetched configuration file “XXX”, from server “XXX”, has an invalid format. No recorded settings applied.

Table 122: Configuration File Fetching Error Handling (Continued)

Error Type	Cause	Behaviour
Empty file	Committing an empty file.	Send a syslog warning message including the file location/name with the transfer server address: The fetched configuration file "XXX", from server "XXX", is empty.
Invalid file content	The file contains invalid characters. Allowed characters are ASCII codes 10 (LF), 13(CR), and 32 to 126.	Send a syslog warning message including the file location/name, the transfer server address and the invalid character (ASCII code): The fetched configuration file "XXX", from server "XXX", has an invalid character "ASCII code XXX". No recorded settings applied.
Invalid transfer server address	The server address is not valid.	Send a syslog warning message including the transfer server address: No configuration file is fetched because the server host "XXX" is invalid. Set <i>sysAdminDownloadConfigFileStatus</i> to <i>fail</i> .
File size too big	Downloading a file with a size exceeding 512000 bytes.	Send a syslog warning message including the file location/name, the transfer server address, the file size and the maximum allowed size: The fetched configuration file "XXX", from server "XXX", has a size "XXX bytes" that exceeds the maximum allowed size "XXX bytes". Set <i>sysAdminDownloadConfigFileStatus</i> to <i>fail</i> .
Invalid encryption	The configuration file cannot be decrypted. A badly encrypted file is detected if the header or the padding is invalid.	Send a syslog warning message including the file location/name and the transfer server address: The fetched configuration file \"%s\", from server \"%s\", can not be decrypted.
TFTP-Specific Error Handling		
File not found	Received error code 1 (file not found) from the TFTP server.	Send a syslog warning message including the file name and location with the TFTP server address: The configuration file "XXX" was not found on the TFTP server "XXX". Set <i>sysAdminDownloadConfigFileStatus</i> to <i>fail</i> .
Access violation	Received error code 2 (access violation) from the TFTP server.	Send a syslog warning message including the file name and location with the TFTP server address: The configuration file "XXX" was not fetched. There was a TFTP access violation with server "XXX". Set <i>sysAdminDownloadConfigFileStatus</i> to <i>fail</i> .

Table 122: Configuration File Fetching Error Handling (Continued)

Error Type	Cause	Behaviour
Connection timeout	No answer from the TFTP server. The time elapsed since the TFTP request was sent exceeds 32 seconds.	Send a syslog warning message including the file name and location with the TFTP server address: The configuration file "XXX" was not fetched. The TFTP connection with server "XXX" timed out. Set <i>sysAdminDownloadConfigFileStatus</i> to <i>fail</i> .
Transfer error	Received a TFTP error (other than error code 1 and 2) from the TFTP server.	Send a syslog warning message including the file name and location with the TFTP server address: Error in the TFTP transfer of the configuration file "XXX" from host "XXX" and port number XXX. Set <i>sysAdminDownloadConfigFileStatus</i> to <i>fail</i> .
File size too big	Downloading a file with a size exceeding 512000 bytes.	Abort the transfer by sending error code 3 (disk full or allocation exceeded) to the TFTP client.
HTTP/HTTPS-Specific Error Handling		
Access unauthorized	Received a 401 Unauthorized from the HTTP server.	Send a syslog warning message including the file location/name with the HTTP server address: The access to configuration file "XXX" is unauthorized on HTTP server "XXX". Set <i>sysAdminDownloadConfigFileStatus</i> to <i>fail</i> .
File not found	Received a 404 Not Found from the HTTP server.	Send a syslog warning message including the file location/name with the HTTP server address: The configuration file "XXX" was not found on the HTTP server "XXX". Set <i>sysAdminDownloadConfigFileStatus</i> to <i>fail</i> .
Session timeout	No answer from the HTTP server. The time elapsed since the HTTP request was sent exceeds 15 seconds.	Send a syslog warning message including the file location/name with the HTTP server address: The configuration file "XXX" was not fetched. The HTTP session with server "XXX" timed out. Set <i>sysAdminDownloadConfigFileStatus</i> to <i>fail</i> .
Session closed by peer	The HTTP server closed the session.	Send a syslog warning message including the file location/name with the HTTP server address: The configuration file "XXX" HTTP transfer session was closed by peer: host "XXX". Set <i>sysAdminDownloadConfigFileStatus</i> to <i>fail</i> .

Table 122: Configuration File Fetching Error Handling (Continued)

Error Type	Cause	Behaviour
Transfer error	Received an HTTP error (other than 401 and 404) from the HTTP server.	Send a syslog warning message including the file location/name with the HTTP server address and port: Error in the HTTP transfer of the configuration file "XXX" from host "XXX" and port number XXX. Set <code>sysAdminDownloadConfigFileStatus</code> to <i>fail</i> .

Management Server

You can set the Mediatrix 4100 so that it asks the management server to send it a configuration file.



Note: Downloading a configuration file from the management server can only be performed through the TFTP protocol.

Management Server Configuration

To download a configuration file from the management server, you must setup the management server information as per ["Chapter 26 - Management Server Configuration" on page 335](#).

Downloading from the Management Server

Once the management server has been properly set up, you can define the configuration file download.



In the *Unit Manager Network Administration Manual*, refer to chapter *Performing Actions on Mediatrix Units*, section *Downloading a Configuration File*.

► To download the configuration file from the management server:

1. Place the configuration file on the computer hosting the management server.
2. In the `sysConfigMIB`, request a configuration file download by setting the `sysConfigDownloadConfigFile` variable to **requestFileDownload**.
3. Set the `sysConfigDownloadConfigMode` variable to **request**.

The Mediatrix 4100 sends a notification, *msTrapConfigInformation*, to the management server, via SNMP traps, to request the configuration file.

The management server then initiates the TFTP session and pushes the file into the unit.

If the management server is the Unit Manager Network from Media5, the following steps are automatically performed. If you are using another management server, you may have to perform them manually.

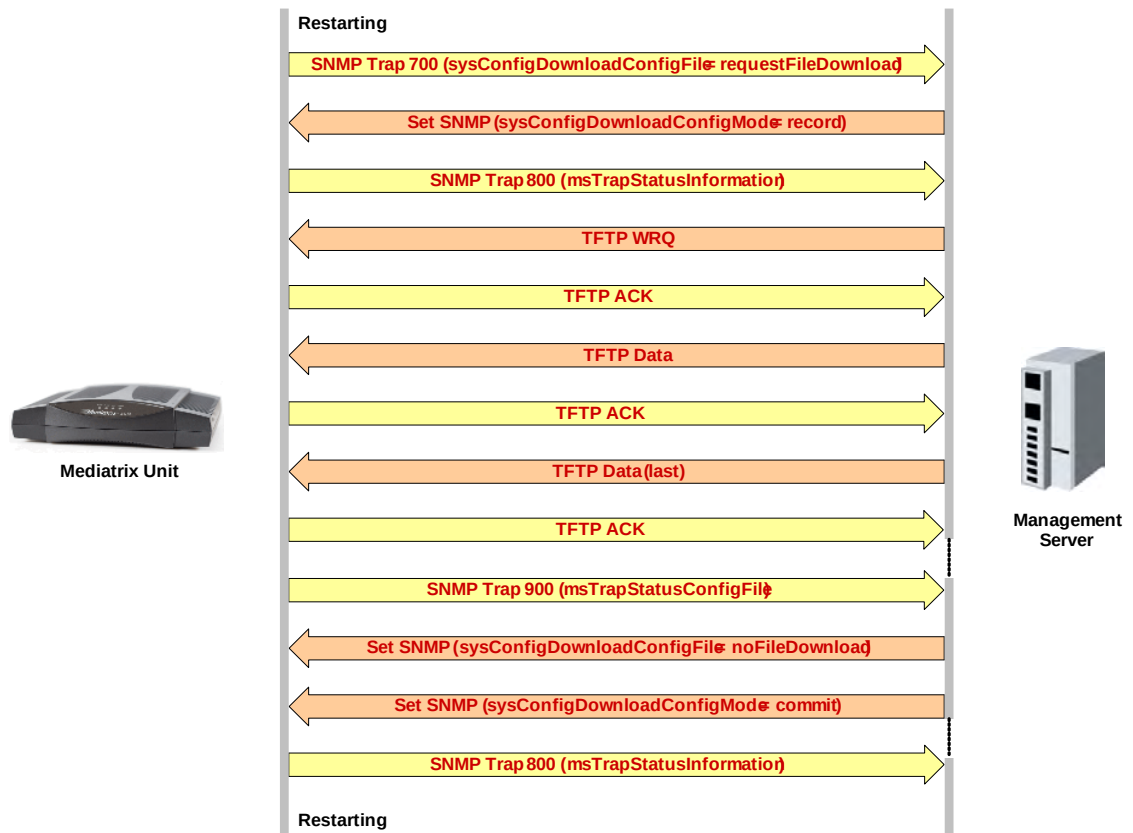
- a. The Unit Manager Network sets the `sysConfigDownloadConfigMode` variable to **record**.
- b. The Unit Manager Network sends the configuration file to the Mediatrix 4100.
- c. Once the configuration file has been sent, the Unit Manager Network sets the `sysConfigDownloadConfigFile` variable to **noFileDownload**.
- d. The Unit Manager Network sets the `sysConfigDownloadConfigMode` variable to **commit**.

If a valid configuration file is successfully downloaded, then the Mediatrix 4100 automatically restarts to apply all the new settings. If the Mediatrix 4100 does not restart, this could mean the download failed. In this case, you can query the status of the last configuration file download in the `sysAdminDownloadConfigFileStatus` variable:

- idle: No configuration file download has been performed yet.

- fail: The last configuration file download failed.
- success: The last configuration file download succeeded.
- inProgress: A configuration file download is in progress.
- listening: The unit is listening and waiting for a configuration file to be sent by the management server.

Figure 80: Configuration Sequence Update Using the Management Server



Error Handling

The following are possible error sources when updating the unit configuration using the management server. The error cause and the unit behaviour are also described.

Table 123: Configuration File Error Handling with the Management Server

Error Type	Cause	Behaviour
Empty file	Committing an empty file.	Send a syslog warning message including the file name and the TFTP client address: The configuration file "XXX" pushed to the unit by the TFTP client "XXX" is empty.
Invalid file content	Committing a file that contains invalid characters. Allowed characters are ASCII codes 10 (LF), 13(CR), and 32 to 126.	Send a syslog warning message including the file name, the TFTP client address and the invalid character (ASCII code): The configuration file "XXX" pushed to the unit by the TFTP client "XXX" has an invalid character "ASCII code XXX". No recorded settings applied.

Table 123: Configuration File Error Handling with the Management Server (Continued)

Error Type	Cause	Behaviour
Invalid file format	Committing a file with an invalid format.	Send a syslog warning message including the file name and the TFTP client address: The configuration file "XXX" pushed to the unit by the TFTP client "XXX" has an invalid format. No recorded settings applied.
File size too big	Downloading a file with a size exceeding 512000 bytes.	Send a syslog warning message including the file name, the TFTP client address, the file size and the maximum allowed size: The configuration file "XXX" from the TFTP client "XXX" is not downloaded because its size "XXX bytes" exceeds the maximum allowed size "XXX bytes". Send error code 3 (disk full or allocation exceeded) to the TFTP client. Set <i>sysAdminDownloadConfigFileStatus</i> to <i>fail</i> and send <i>msTrapStatusConfigFile</i> .
TFTP transfer error	Received a TFTP error from the TFTP client.	Send a syslog warning message including the file name and the TFTP client address: Error in the TFTP transfer of the configuration file "XXX" from the TFTP client "XXX". Set <i>sysAdminDownloadConfigFileStatus</i> to <i>fail</i> and send <i>msTrapStatusConfigFile</i> .
TFTP transfer aborted	The transfer was aborted while in progress by changing the value of <i>sysConfigDownloadConfigMode</i> or <i>sysConfigDownloadConfigFile</i> .	Send a syslog warning message including the file name and the TFTP client address: The TFTP transfer of the configuration file "XXX" from the TFTP client "XXX" was aborted. Set <i>sysAdminDownloadConfigFileStatus</i> to <i>fail</i> and send <i>msTrapStatusConfigFile</i> .
File pulling not allowed	A TFTP client is trying to read a file from the unit.	Send a syslog informational message including the file name and the TFTP client address: The TFTP client "XXX" is trying to pull the file "XXX" from the unit. This is not allowed. Send error code 2 (access violation) to the TFTP client.

Syslog Messages

A syslog message is sent whenever it is impossible for the management server to download a configuration file or when it is impossible to apply the new settings to the unit.

Table 124: Syslog Messages Using the Management Server

Level	Message	Event
Warning	The notification "XXX" could not be sent to msHost "XXX" and msTrapPort XXX.	A SNMP trap could not be sent to the management server. The syslog warning message includes the SNMP trap number, the management server address and port.

Table 124: Syslog Messages Using the Management Server (Continued)

Level	Message	Event
Informational	Parameter values defined in the configuration file were successfully committed. Restarting the unit...	A downloaded configuration file was successfully committed.
Warning	None of the parameter values defined in the configuration file was successfully committed.	No parameter value from the downloaded configuration file was successfully applied (e.g., because of bad OIDs).

Configuration File Example

The configuration file format uses XML (eXtensible Markup Language). The following is the accepted format:

```
<MX_Config_File FileId="MX_MIBFILE" MIBVersionNumber="" VersionNumber="1.0">
  <Object Prefix="" Suffix="" Value=""/>
  <Object Prefix="" Suffix="" Value=""/>
</MX_Config_File>
```

The following is an example of a configuration file:

```
<MX_Config_File FileId="MX_MIBFILE" MIBVersionNumber="1.0" VersionNumber="1.0">
  <Object Prefix="1.3.6.1.4.1.4935.15.1.8.1" Suffix="0" Value="0"/>
  <Object Prefix="1.3.6.1.4.1.4935.15.1.10.1" Suffix="0" Value="0"/>
  <Object Prefix="1.3.6.1.4.1.4935.15.1.10.10.1" Suffix="0" Value="192.168.0.10"/>
  <Object Prefix="1.3.6.1.4.1.4935.15.1.20.1" Suffix="0" Value="0"/>
  <Object Prefix="1.3.6.1.4.1.4935.15.3.1.30.1" Suffix="0" Value="0"/>
  <Object Prefix="1.3.6.1.4.1.4935.15.3.1.30.3" Suffix="0" Value="ConfigFile.xml"/>
  <Object Prefix="1.3.6.1.4.1.4935.15.15.1.5" Suffix="0" Value="0"/>
  <Object Prefix="1.3.6.1.4.1.4935.15.35.1.20.1.1" Suffix="3" Value="0"/>
  <Object Prefix="1.3.6.1.4.1.4935.15.35.1.20.1.1" Suffix="4" Value="0"/>
</MX_Config_File>
```

Supported Characters

When creating and/or editing a configuration file, the following ASCII codes are supported:

10 LF, line feed	62 >, greater than	94 ^, caret
13 CR, carriage return	63 ?, question mark	95 _, underscore
32 space	64 @, commercial at	96 `, back quote
33 !, exclamation mark	65 A	97 a
34 ", double quote	66 B	98 b
35 #, hash	67 C	99 c
36 \$, dollar	68 D	100 d
37 %, percent	69 E	101 e
38 &, ampersand	70 F	102 f
39 ', quote	71 G	103 g
40 (, open parenthesis	72 H	104 h
41), close parenthesis	73 I	105 i
42 *, asterisk	74 J	106 j
43 +, plus	75 K	107 k
44 ,, comma	76 L	108 l
45 -, minus	77 M	109 m
46 ., full stop	78 N	110 n
47 /, oblique stroke	79 O	111 o
48 0, zero	80 P	112 p
49 1	81 Q	113 q
50 2	82 R	114 r
51 3	83 S	115 s
52 4	84 T	116 t
53 5	85 U	117 u
54 6	86 V	118 v
55 7	87 W	119 w
56 8	88 X	120 x
57 9	89 Y	121 y
58 :, colon	90 Z	122 z
59 ;, semicolon	91 [, open square bracket	123 {, open curly bracket
60 <, less than	92 \, backslash	124 , vertical bar
61 =, equals	93], close square bracket	125 }, close curly bracket
		126 ~, tilde

All other ASCII codes will result in an invalid configuration file.

This chapter describes how to download a software version available on the designated software server into the Mediatrix 4100.

You have the choice to perform the software download by using the TFTP, HTTPS, or HTTP protocol. You can also configure the Mediatrix 4100 to automatically update its software version.

You can also set these parameters via the web interface, as described in [“Firmware Download” on page 51](#).

Before Downloading

To download a software, you may need to setup the following applications on your computer:

- ▶ TFTP server with proper root path
- ▶ MIB browser (with the current Mediatrix 4100 MIB tree)
You can use the MIB browser built in the Media5's Unit Manager Network. See [“Unit Manager Network – Element Management System” on page xxii](#) for more details.
- ▶ Software upgrade zip file
- ▶ SNTP server properly configured
- ▶ HTTP server with proper root path
- ▶ HTTPS server with proper root path
- ▶ Syslog daemon (optional)

Configuring the TFTP Server

If you are to perform a software download by using the TFTP protocol, you must install a TFTP (Trivial File Transfer Protocol) server running on the PC designated as the software file server. This PC must not have a firewall running. Media5 also recommends to place the PC and the Mediatrix 4100 in the same subnet.

It is assumed that you know how to set the TFTP root path. If not, refer to your TFTP server's documentation.

Configuring the SNTP Server

If you are to use the automatic software update feature (see [“Automatic Software Update” on page 217](#) for more details) or the HTTPS protocol, you must have a time server SNTP that is accessible and properly configured. It is assumed that you know how to configure your SNTP server. If not, refer to your SNTP server's documentation. You can also refer to [“Chapter 20 - SNTP Settings” on page 289](#) for more details on how to configure the Mediatrix 4100 for a SNTP server.

Configuring the HTTP Server

If you are to perform a software download by using the HTTP protocol, you must install a HTTP server running on the PC designated as the server host. This PC must not have a firewall running. Media5 also recommends to place the PC and the Mediatrix 4100 in the same subnet.

It is assumed that you know how to set the root path. If not, refer to your HTTP server's documentation.

Configuring the HTTPS Server

Standards Supported

- RFC 2246 – The TLS Protocol Version 1.0
- RFC 2818 – HTTP Over TLS
- RFC 2459 – X.509 Digital Certificates
- RFC 3280 – Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile

If you are to perform a software download that requires authentication or privacy by using the HTTP over the Transport Layer Security (TLS) protocol (HTTPS), you must install a HTTPS server running on the PC designated as the server host. It is assumed that you know how to set the root path and set the SSL/TLS security configuration. If not, refer to your HTTPS server's documentation.



Caution: You must have a time server SNTP that is accessible and properly configured, or the automatic software update feature may not work properly. It is assumed that you know how to configure your SNTP server. If not, refer to your SNTP server's documentation. You can also refer to [“Chapter 20 - SNTP Settings” on page 289](#) for more details on how to configure the Mediatrix 4100 for a SNTP server.

The Mediatrix 4100 supports the following SSL/TLS parameters:

Table 125: Secure Parameters Supported

Supported Parameter	Description
Key Exchange Mechanism	• RSA • Diffie-Hellman
Ciphers	• AES (128 and 256 bits) • 3DES (168 bits)
Message Digests	• SHA-1



Note: Media5 recommends to use cipher suites based on the RSA key exchange mechanism, because the Diffie-Hellman key exchange mechanism introduces a noticeable delay in the HTTPS session establishment.

Certificates

The Mediatrix 4100 contains embedded security certificates formatted as per ITU x.509 and RFC 3280. The certificates are factory-installed. The name of X.509 certificates currently installed in the Mediatrix 4100 are listed in the *securityCertificatesMIB* under the *certificateName* variable (under the *certificateTable* group). You must have at least one matching certificate on your HTTPS server.

You can also find the following information:

Table 126: Certificates

Parameter	Description
certificateSubjectCommonName	The certificate's subject name. If the certificate contains a subject field, display the common name. Otherwise display the first non-empty subject alternative name from the following list: • Uniform Resource Locator • DNS name • IP Address • RFC 822 name See RFC 3280 sections 4.1.2.6 and 4.2.1.7 for details.

Table 126: Certificates (Continued)

Parameter	Description
certificateExpirationDate	The certificate's expiration date. Display the date at which the certificate expires. The format is MM/DD/YYYY in universal time. See RFC 3280 section 4.1.2.5 for details.

When contacting a HTTPS server, the Mediatrix 4100 establishes a TLS connection by (among others):

- ▶ negotiating cipher suites
- ▶ checking the server certificates validity (dates)

The Mediatrix 4100 then checks the server's identity by validating the host name used to contact it against the information found in the server's certificate, as described in RFC 2818, section 3.1.

If any of the above does not succeed, the Mediatrix 4100 refuses the secure connection. To help detect such errors, you can increase the syslog messages level.

Software Servers Configuration

The Mediatrix 4100 must know the IP address and port number of its Primary and Secondary software servers. These servers contain the files required for the software update. You can assign these information to the Mediatrix 4100 through a DHCP server or manually enter them yourself in static variables.



In the *Unit Manager Network Administration Manual*, refer to chapter *Administration Parameters*, section *Software and Emergency Download*.

DHCP Configuration

Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See ["Chapter 8 - IP Address and Network Configuration" on page 143](#) for more details.

▶ To use DHCP-assigned information:

1. In the *ipAddressConfig* folder, locate the *imageSelectConfigSource* variable (under the *ipAddressConfigImage* group).
This variable defines whether the Mediatrix 4100 must ask for its Image server settings through a DHCP server or not.
2. Set the *imageSelectConfigSource* variable to **dhcp**.
You can query the Image server's IP address and port number assigned by the DHCP server in the following read-only variables (in the *ipAddressStatus* folder):
 - imagePrimaryHost
 - imagePrimaryPort
 - imageSecondaryHost
 - imageSecondaryPort
3. Set how you want to define the Primary Image server information in the DHCP server.

Table 127: Primary Image Server DHCP Information

To use a...	Set...
vendor specific code	The <i>imageDhcpPrimarySiteSpecificCode</i> variable to 0 . Set the Primary image server IP address in the DHCP server inside the vendor specific sub-option 117 (hexadecimal 0x75).

Table 127: Primary Image Server DHCP Information (Continued)

To use a...	Set...
site specific code	The <i>imageDhcpPrimarySiteSpecificCode</i> variable to any value between 128 and 254. Set the Primary image server IP address in the DHCP server inside the site specific option you have chosen (it must match the value of the <i>imageDhcpPrimarySiteSpecificCode</i> variable in the unit's configuration).

See ["Vendor and Site Specific DHCP Options" on page 156](#) for more details.

4. Set how you want to define the Secondary Image server information in the DHCP server.

Table 128: Secondary Image Server DHCP Information

To use a...	Set...
vendor specific code	The <i>imageDhcpSecondarySiteSpecificCode</i> variable to 0 . Set the Secondary image server IP address in the DHCP server inside the vendor specific sub-option 118 (hexadecimal 0x76).
site specific code	The <i>imageDhcpSecondarySiteSpecificCode</i> variable to any value between 128 and 254. Set the Secondary image server IP address in the DHCP server inside the site specific option you have chosen (it must match the value of the <i>imageDhcpPrimarySiteSpecificCode</i> variable in the unit's configuration).

See ["Vendor and Site Specific DHCP Options" on page 156](#) for more details.

Static Configuration

Use the static configuration if you are not using a DHCP server or if you want to bypass it.

► To use static information:

1. In the *ipAddressConfig* folder, locate the *imageSelectConfigSource* variable.
This variable defines whether the Mediatrix 4100 must ask for its Image server settings through a DHCP server or not.
2. Set the *imageSelectConfigSource* variable to **static**.
3. Set the following variables:

Table 129: Image Static Information

Variable	Description
imageStaticPrimaryHost	Static primary image server IP address or domain name. This is the current address of the PC that hosts the files required for the download (extracted from the zip file). Default Value: 192.168.0.10
imageStaticPrimaryPort	Static primary image server IP port number. Default Value: 69
imageStaticSecondary Host	Static secondary image server IP address or domain name. This is the current address of the PC that hosts the files required for the download (extracted from the zip file). Default Value: 192.168.0.10
imageStaticSecondaryPort	Static secondary image server IP port number. Default Value: 69

The default port value complies to RFC 1340 on the well-known ports (assigned numbers). This value (69) applies to a TFTP server. It may be different for other servers. If you are using an HTTP/HTTPS server, you must change the port value to 80.

Download Procedure

The following describes how to download a software version into the Mediatrix 4100.



Note: Configuration settings are not lost when upgrading the software to a newer version. However, configuration settings may be lost if you upload an older firmware to the device. See [“Software Downgrade” on page 220](#) for more details.

You have the choice to perform the software download by using the TFTP, or HTTP, or HTTPS protocol. You can also configure the Mediatrix 4100 to automatically update its software version.



In the *Unit Manager Network Administration Manual*, refer to chapter *Administration Parameters*, section *Software and Emergency Download*.

Extracting the Zip File

The zip file contains the software information required for the download.

Extract the contents of the zip file on the PC designated as the software file server. Be sure to use the defined folder name. This creates a directory that contains the files required for the Mediatrix 4100 to properly update its software.

The directory name must be the same as the name defined in the *imageLocation* or *imageSelectionFileLocation* variable of the *imageMIB*. See [“Setting up the Image Path” on page 211](#) for more details.

Media5 suggests that a folder, named identically to the software build, be available and used for the files related to that build only. Each folder should include only one delivery to ensure accuracy.

This directory must be located under the root path as defined in the TFTP/HTTP/HTTPS server or the software download will not proceed.

Setting up the Image Path

When performing a software download, you must configure the path, on the remote image server, of the directory where you extracted the files required for the download. This applies to both the manual or automatic download procedure, using the HTTP, HTTPS, or TFTP protocol.

The directory must be located under the root path, as defined in the TFTP, or HTTP, or HTTPS server, or the software download will not proceed. See [“Before Downloading” on page 207](#) for more details.

The Mediatrix 4100 first downloads a file called “setup.inf”. This file contains the list of all the other files to download, depending on the product. The “setup.inf” file and all the other files must be in the same directory. If any of the files is missing, the procedure will not work properly.

► To setup the Image path:

1. In the *imageMIB*, select where to get the image location in the *imageLocationProvisionSource* variable.

You have the following choices:

Table 130: Image Location Parameters

Parameter	Description
static	Uses the directory specified in the <i>imageLocation</i> variable (see Step 2).

Table 130: Image Location Parameters (Continued)

Parameter	Description
remoteFile	The image location is defined in a file called "mediatrix4108targetimage.inf" (the 4108 part may be changed to 4116 or 4124 depending on your product). The location of this file is defined in the <i>imageSelectionFileLocation</i> variable. This is useful if you are using automatic updates with multiple units (see Step 3).

2. If you have set the *imageLocationProvisionSource* variable to **static** (see Step 1), configure the path in the *imageLocation* variable.
This is the location of the "setup.inf" file that contains the list of the files to download into the Mediatrix 4100. The "setup.inf" file and all the other files must be in the same directory. In other words, this is the path from the root TFTP/HTTP/HTTPS folder down to the files extracted from the zip file.
Note that the path must contain a maximum of 63 characters.
3. If you have set the *imageLocationProvisionSource* variable to **remoteFile** (see Step 1):
 - a. Create a text file and write the path and/or name of the directory that contains the files required for download. Save this file as "mediatrix4108targetimage.inf" under the server root path.



Note: If you leave the file empty, the Mediatrix 4100 will look for the software download information in the root directory of the image server.

- b. Configure the path of the "mediatrix4108targetimage.inf" file in the *imageSelectionFileLocation* variable.
Note that the selection file name is in lower case. Some web servers are case sensitive. The path must contain a maximum of 63 characters.
This is useful if you are using automatic updates with multiple units. If you want the units to download a new version, you only have to change the path once in the "mediatrix4108targetimage.inf" file. If you were to use the *imageLocation* variable, you would have to change the path in every unit.

Let's consider the following example:

- The directory that contains the files required for download is called: **SIP_v5.0.1.1_MX-S5001-01**.
- This directory is under **C:/Root/Download**.

Table 131: Path Configurations Example

Root Path	Corresponding Path Name
c:/root/download	SIP_v5.0.1.1_MX-S5001-01
c:/	root/download/SIP_v5.0.1.1_MX-S5001-01
c:/root	download/SIP_v5.0.1.1_MX-S5001-01

The following are some tips to help your download process:

- Use the "/" character when defining the path to indicate sub-directories. For instance, *root/download*.
If you are using the TFTP protocol to download the software, note that some TFTP servers on Windows do not recognize the "/" character and produce an error. In this case, use the "\" character.
- Use basic directory names, without spaces or special characters such as "~", "@", etc., which may cause problems.
- Cut and paste the path and/or name of the directory that contains the extracted files into the image path of the Mediatrix 4100 (you may have to convert "\" into "/") to eliminate typographical errors.

Note that you can define the **C:/Root/Download** part as you want. The file names may also differ from the example shown above.

Software Download Status

You can validate the status of the software download in various ways.

Syslog Messages

If you are using a Syslog daemon, you will receive messages that inform you of the software update status. The following are the syslog messages the Mediatrix 4100 sends:

Table 132: Software Update Syslog Messages

Level	Message	Event
General Messages		
Informational	The software update succeeded.	The software update has been successful.
Error	The software update failed.	The software update experienced an error and has not been completed.
Error	The software update failed (xxx).	An error occurs when updating the software, internal error code provided.
Warning	Primary image server not specified, cannot download file: xxx	This error occurs when an image download is initiated and no domain name or address is specified for the primary image server.
Warning	Secondary image server not specified, cannot download file: xxx.	When a request involving the primary server fails, the secondary server is tried. This error occurs when there is no address or domain name specified for the secondary image server.
Error	Cannot resolve address of image server: xxx.	A DNS request failed to resolve the domain name of the image server (primary or secondary).
Error	Target image at location: xxx from host: xxx is invalid or corrupted.	For periodic and automatic updates, the target image to download is first compared with the installed image. This error occurs when this comparison failed because of corruption in the target image files.
Informational	Image download transfer initiated.	When manual, periodic or "at restart" image download is initiated.
Warning	The file: xxx from host: xxx exceeds the size limit.	The selection file or "setup.inf" file received exceeds 10000 bytes.
Informational	Target image at location: xxx from host xxx is identical to currently installed image. Transfer aborted.	For periodic and automatic updates, the target image to download is first compared with the installed image. This message occurs when this comparison determined that the target image is identical to the installed image.
Error	Image does not support hardware (error %d)	The software download failed because the software image is not compatible with the hardware.

Table 132: Software Update Syslog Messages (Continued)

Level	Message	Event
HTTP-Specific Messages		
Warning	HTTP image transfer of file: xxx from host: xxx was closed by peer.	The HTTP transfer was closed by the peer.
Warning	HTTP image transfer of file: xxx from host: xxx was closed due to unsupported or malformed response from the host.	In the HTTP response, one of the following error occurred: • The protocol version is not 1.0 or 1.1. • Some field or line is not properly formatted. • The trailing <crLf> is not present at the end of the header. • Unsupported kind of response.
Warning	HTTP image transfer of file: xxx from host: xxx was refused because of a malformed or incompatible request.	When receiving HTTP response #400 or #403.
Warning	HTTP image transfer of file: xxx from host: xxx was refused because of a server error.	When receiving HTTP response #500 or #501.
Warning	HTTP image transfer of file: xxx from host: xxx was refused because service is unavailable.	When receiving HTTP response #503.
TFTP-Specific Messages		
Warning	Image transfer of file: xxx from host: xxx and port: xxx was closed due to unexpected error	Unexpected error, either internal or on a TFTP or HTTP connection.
Warning	Image transfer of file: xxx from host: xxx port: xxx was closed after timeout	When not receiving TFTP packets for 32 seconds or not receiving a HTTP packet for 15 seconds.
Warning	Image transfer. File: xxx not found on host: xxx	When receiving TFTP error "NOT FOUND" or HTTP response #404.
Warning	Image transfer. Access to file: xxx on host: xxx is unauthorized	When receiving TFTP error "ACCESS" or HTTP response #401.

LED States

When the Mediatrix 4100 initiates a software download, the LEDs located on the front panel indicate the status of the process.

Table 133: LED States in Software Download

Event	LED State
Image downloading and writing	<i>Power, LAN, In Use</i> and <i>Ready</i> LEDs blink alternately at 1 Hz with 1/4 ON duty cycle. Warning: Do not turn the Mediatrix 4100 off while in this state.
Image download failed	<i>Power, LAN, In Use</i> and <i>Ready</i> LEDs blink at the same time at 2 Hz with 50% ON duty cycle for 4 seconds.

See "[LED Indicators](#)" on page 15 for a detailed description of the LED patterns related to the software download process.

MIB Variable

You can validate the result of the last software update by checking the state of the `sysAdminLastDownloadSoftware` MIB variable.

Download via TFTP

The following steps explain how to download a software by using the TFTP protocol.



In the *Unit Manager Network Administration Manual*, refer to chapter *Performing Actions on Mediatrix Units*, section *Downloading a Software Version*.

► To download a software via TFTP:

1. If not already done, setup the Image server used to download the software (see "[Before Downloading](#)" on page 207).
2. Be sure that UDP ports 60000 to 60512 inclusively are opened in your firewall.
3. If not already done, configure the Image path as described in "[Setting up the Image Path](#)" on page 211.
4. If not already done, configure the image hosts and ports as defined in "[Software Servers Configuration](#)" on page 209.
5. Set the TFTP root path in your TFTP server.
It is assumed that you know how to set the TFTP root path. If not, refer to your TFTP server's documentation.
6. Set the `imageTransferProtocol` variable to **tftp**.
7. Set the `groupSetAdmin` variable (in the `groupAdminMIB`) to **ForceLock**.
All activities in progress on the Mediatrix 4100 are terminated immediately and the unit enters the maintenance mode (the value of the `groupAdminState` variable is "locked"). The software upgrade may take place.
The Mediatrix 4100 lines will be unlocked after successfully downloading the software and restarting. If, for any reason, the software download is not successful, you must manually unlock the lines as per "[Lines Administrative State](#)" on page 223.
8. Initiate the download by setting the `sysAdminCommand` variable (in the `sysAdminMIB`) to **downloadSoftware**.
This starts the download process.



Caution: Never shutdown the Mediatrix 4100 manually while in the download process, because the image may be partially written and the Mediatrix 4100 is unable to restart.

The software download may take several minutes, depending on your Internet connection, network conditions and servers conditions.

9. Update the MIB browser with the MIB version coming with the software version.

Download via HTTP/HTTPS

The following steps explain how to download a software by using the HTTP protocol.

- or HTTPS If you are using HTTPS, the Mediatrix 4100 must contain the proper certificate. See [“Configuring the HTTPS Server” on page 208](#) for more details. **To download a software via HTTP or HTTPS:**

1. If not already done, setup the Image server used to download the software (see [“Before Downloading” on page 207](#)).
2. If not already done, configure the Image path as described in [“Setting up the Image Path” on page 211](#).
3. If not already done, configure the image hosts and ports as defined in [“Software Servers Configuration” on page 209](#).



Caution: When downloading via HTTP or HTTPS, the image server's port must be 80. You can query the actual port assigned in the *imagePrimaryPort* and *imageSecondaryPort* read-only variables (in the *ipAddressStatus* folder).

If you are using a DHCP server and it did not provide the proper port, reconfigure it with the proper port or use a static configuration. See [“Software Servers Configuration” on page 209](#) for more details.

4. In the *imageMIB*, set the *imageTransferProtocol* variable to **http** or **https**.
Your HTTP or HTTPS server may activate some caching mechanism for the software download. This mechanism caches the initial software download for later processing, thus preventing changes or update of the original download by the user. This can cause problems if a user wants to modify the software download and perform it again immediately. The result will still return the original download and not the new one.
5. If your HTTP or HTTPS server requires authentication, set the following:
 - The user name in the *imageTransferUsername* variable.
 - The password in the *imageTransferPassword* variable.
6. Set the *groupSetAdmin* variable (in the *groupAdminMIB*) to **ForceLock**.
All activities in progress on the Mediatrix 4100 are terminated immediately and the unit enters the maintenance mode (the value of the *groupAdminState* variable is “locked”). The software upgrade may take place.
The Mediatrix 4100 lines will be unlocked after successfully downloading the software and restarting.
If, for any reason, the software download is not successful, you must manually unlock the lines as per [“Lines Administrative State” on page 223](#).
7. Initiate the download by setting the *sysAdminCommand* variable (in the *sysAdminMIB*) to **downloadSoftware**.
This starts the download process.



Caution: Never shutdown the Mediatrix 4100 manually while in the download process, because the image may be partially written and the Mediatrix 4100 is unable to restart.

The software download may take several minutes, depending on your Internet connection, network conditions and servers conditions.

8. Update the MIB browser with the MIB version coming with the software version.

User Agent Header of HTTP Requests

The *User-Agent* header field of an HTTP request contains information about the user agent client originating the request. For instance, the information of the *User-Agent* header could be something like the following:

User-Agent: Softphone Beta1.5

You can customize the information that the Mediatrix 4100 sends when establishing a communication.

► **To customize the HTTP User Agent header of HTTP requests:**

1. In the *interopMIB*, set the *mxInteropHttpUAHeaderConfig* variable with the proper macro.
The following macros are replaced by their representation:
 - **%version%**: Version of the application.
 - **%mac%**: Unit MAC address (lowercase).
 - **%rev%**: Hardware revision number.
 - **%product%**: Product name.
 - **%%**: A '%' sign.
2. Restart the Mediatix 4100 so that the changes may take effect.

Automatic Software Update

You can configure the Mediatix 4100 to automatically update its software version. This update can be done:

- Every time the Mediatix 4100 restarts.
- At a specific time interval you can define.

Automatic Update on Restart

The Mediatix 4100 may download a new software version each time it restarts.

► **To set the automatic update every time the Mediatix 4100 restarts:**

1. If not already done, setup the Image server used to download the software (see [“Before Downloading” on page 207](#)).
2. If not already done, configure the Image path as described in [“Setting up the Image Path” on page 211](#).
3. If not already done, configure the image hosts and ports as defined in [“Software Servers Configuration” on page 209](#).



Caution: When downloading via HTTP or HTTPS, the image server's port must be 80. You can query the actual port assigned in the *imagePrimaryPort* and *imageSecondaryPort* read-only variables (in the *ipAddressStatus* folder).

If you are using a DHCP server and it did not provide the proper port, reconfigure it with the proper port or use a static configuration. See [“Software Servers Configuration” on page 209](#) for more details.

4. In the *imageMIB*, set the *imageTransferProtocol* variable to either **http**, **https**, or **fttp**.
If you are using the HTTPS protocol, the Mediatix 4100 must contain the proper certificate. See [“Configuring the HTTPS Server” on page 208](#) for more details.
If you are using the HTTP or HTTPS protocol to download the software, be aware that your HTTP or HTTPS server may activate some caching mechanism for the software download. This mechanism caches the initial software download for later processing, thus preventing changes or update of the original download by the user. This can cause problems if a user wants to modify the software download and perform it again immediately. The result will still return the original download and not the new one.
5. If you are using the HTTP or HTTPS protocol and your HTTP or HTTPS server requires authentication, set the following:
 - The user name in the *imageTransferUsername* variable.
 - The password in the *imageTransferPassword* variable.
6. Set the *imageAutoUpdateOnRestartEnable* variable to **enable**.
7. Set the *imageAutoUpdateEnable* variable to **enable**.
The automatic software update will be performed each time the Mediatix 4100 restarts.

Automatic Update at a Specific Time Interval

You can configure the Mediatrix 4100 to download a software version at a specific day and/or time.

► To set the automatic update at a specific time interval:

1. If not already done, setup the Image server used to download the software (see [“Before Downloading” on page 207](#)).
2. If not already done, configure the Image path as described in [“Setting up the Image Path” on page 211](#).
3. If not already done, configure the image hosts and ports as defined in [“Software Servers Configuration” on page 209](#).



Caution: When downloading via HTTP or HTTPS, the image server's port must be 80. You can query the actual port assigned in the *imagePrimaryPort* and *imageSecondaryPort* read-only variables (in the *ipAddressStatus* folder).

If you are using a DHCP server and it did not provide the proper port, reconfigure it with the proper port or use a static configuration. See [“Software Servers Configuration” on page 209](#) for more details.

4. In the *imageMIB*, set the *imageTransferProtocol* variable to either **http**, **https**, or **tftp**.
If you are using the HTTPS protocol, the Mediatrix 4100 must contain the proper certificate. See [“Configuring the HTTPS Server” on page 208](#) for more details.
If you are using the HTTP or HTTPS protocol to download the software, be aware that your HTTP or HTTPS server may activate some caching mechanism for the software download. This mechanism caches the initial software download for later processing, thus preventing changes or update of the original download by the user. This can cause problems if a user wants to modify the software download and perform it again immediately. The result will still return the original download and not the new one.
5. If you are using the HTTP or HTTPS protocol and your HTTP or HTTPS server requires authentication, set the following:
 - The user name in the *imageTransferUsername* variable.
 - The password in the *imageTransferPassword* variable.
6. Define the time base for automatic software updates in the *imageAutoUpdateTimeUnit* variable (in the *imageAutomaticUpdate* group).
You have the following choices:

Table 134: Time Unit Parameters

Parameter	Description
minutes	Updates the software every x minutes. You can specify the x value in the variable <i>imageAutoUpdatePeriod</i> (see Step 7).
hours	Updates the software every x hours. You can specify the x value in the variable <i>imageAutoUpdatePeriod</i> (see Step 7).
days	Updates the software every x days. You can specify the x value in the variable <i>imageAutoUpdatePeriod</i> (see Step 7). You can also define the time of day when to perform the update in the <i>imageAutoUpdateTimeRange</i> variable (see Step 8).

7. Set the waiting period between each software update in the *imageAutoUpdatePeriod* variable. The time unit for the period is specified by the *imageAutoUpdateTimeUnit* variable (see Step 6). Available values are from 1 to 48.

8. If you have selected **days** in Step 6, set the time of the day when to initiate a software update in the *imageAutoUpdateTimeRange* variable.

The time of the day is based on the *sntpTimeZoneString* variable setting (see [“Chapter 20 - SNTP Settings” on page 289](#) for more details).

You must have a time server SNTP that is accessible and properly configured, or the automatic software update feature may not work properly. It is assumed that you know how to configure your SNTP server. If not, refer to your SNTP server's documentation. You can also refer to [“Chapter 20 - SNTP Settings” on page 289](#) for more details on how to configure the Mediatrix 4100 for a SNTP server.

If a time range is specified, the unit will initiate the image software download at a random time within the interval specified.

The format should be one of the following:

```
hh[:mm[:ss]]
hh[:mm[:ss]] - hh[:mm[:ss]]
```

Where:

hh: Hours.
mm: Minutes.
ss: Seconds.

The image software download is initiated at the first occurrence of this value and thereafter with a period defined by *imageAutoUpdatePeriod*. Let's say for instance the automatic update is set with the time of day at 14h00 and the update period at every 2 days.

- If the automatic update is enabled before 14h00, the first update will take place the same day at 14h00, then the second update two days later at the same hour, and so on.
- If the time range is set to '14:00 - 15:00' and the automatic update is enabled within those hours, the first update will take place the following day. This means that a range of '00:00:00 - 23:59:59' will always take place the next day.

9. Set the *imageAutoUpdateEnable* variable to **enable**.

If one of the telephones/faxes is off-hook, the Mediatrix 4100 will perform the download five minutes after both ports are detected on-hook.

Spanning Tree Protocol (STP)

Many network switches use the Spanning Tree Protocol (STP) to manage Ethernet ports activity. When a software download occurs, the *Ethernet* connector of the Mediatrix 4100 may switch off. This shutdown may trigger these network switches to shutdown the matching Ethernet port for at least one minute. This shutdown on the switch side can prevent software download.

To prevent this, the Mediatrix 4100 supports the STP. However, this management has a potential time cost. It may appear from time to time that software downloads take more time. This is normal.

The following is an example where the STP management impacts the download duration.

- ▶ The software download procedure does not use any DHCP and DNS services.
- ▶ The primary image server is down (or not properly configured).
- ▶ The secondary image server is up and running well.

In this case, the Mediatrix 4100 tries to contact the primary image server. As it is not available, the Mediatrix 4100 retries for two minutes. It contacts the secondary server after that period and starts the software download.



Note: When using the Mediatrix 4100, Media5 recommends to disable the Spanning Tree Protocol on the network to which the unit is connected. See also [“DHCP Options Waiting Time” on page 153](#).

Software Downgrade

It is possible to downgrade a Mediatrix 4100 from the current version (for instance, v5.0rx.x) to an older version (for instance, v4.4rx.x).



Note: If you perform a default reset on the Mediatrix 4100, you must download the current version into the unit before performing the software downgrade procedure.

► **To perform a software downgrade:**

1. Create, in a common folder under the TFTP root path, the current (for instance, v5.0) and older (for instance, v4.4) applications folders.
2. Re-update the Mediatrix 4100 with the current application.
The Mediatrix 4100 runs the current software version (v5.0rx.x).
3. Perform the software downgrade to the older application (v4.4rx.x) as described in ["Download Procedure" on page 211](#).

Emergency Software Procedure

If the software download is suddenly interrupted, it may not be complete. Without any protection against this situation, the Mediatrix 4100 is not functional.

A transfer may be interrupted for the following reasons:

- An electrical shortage.
- The user of the Mediatrix 4100 can accidentally power off the unit.

Depending on the moment when the software download has been interrupted, the emergency software procedure (also called rescue application) can automatically start a new software download to repair the software if it has been corrupted by the interruption. However, there is a small but critical time frame during which unrecoverable errors could happen. This is why it is very important that the unit is not turned off during software downloads.

Using the Emergency Software

When the emergency software procedure starts, the following steps apply:

1. The Mediatrix 4100 tries to initiate the software download with the primary software server.
2. If the software download fails with the primary software server, the Mediatrix 4100 tries to initiate the software download with the secondary software server.
3. If the primary and the secondary servers cannot be reached, the Mediatrix 4100 tries two default servers: 192.168.0.10 and then 192.168.0.2.
If, for some reason, it is impossible to rescue the unit by using the primary and secondary servers, setting up a server at one of these addresses within the correct subnet will provide an ultimate way to rescue the unit. However, if these addresses cannot be reached from the unit's subnet, the default gateway must provide appropriate routing to them.
4. If the software download also fails with the two default servers, the Mediatrix 4100 idles for one minute.
5. After this one minute, the Mediatrix 4100 tries to initiate the software download again.
6. If the software download fails again with the primary, secondary, and default software servers, the Mediatrix 4100 idles for two minutes before attempting to initiate the software download.

7. If the emergency software download still fails, the Mediatrix 4100 tries to initiate the software download again by doubling the delay between each attempt up to a maximum of 16 minutes:
 - first attempt: 1 minute delay
 - second attempt: 2 minutes delay
 - third attempt: 4 minutes delay
 - fourth attempt: 8 minutes delay
 - fifth attempt: 16 minutes delay
 - sixth attempt: 16 minutes delay
 - etc.

This procedure continues until the software download completes successfully. The software download can fail if the software server cannot be reached or if the software directory is not found on the software server.

This chapter describes the features available on the lines connected to the Mediatrix 4100.

For information on voice codecs, see [“Chapter 15 - Voice Transmissions” on page 231](#).

For information on data codecs, see [“Chapter 16 - Fax Transmission” on page 249](#).

Lines Administrative State

You can independently set the administrative state of each analog line of your Mediatrix 4100. This state determines how the Mediatrix 4100 processes calls.

For instance, you must properly unlock the analog lines of the Mediatrix 4100 to properly make and receive calls on all of them.

The administrative states may be applied in two ways:

- ▶ **Temporary:** The administrative state is applied immediately, but it is not kept after the Mediatrix 4100 restarts.
- ▶ **Permanent:** When the Mediatrix 4100 restarts, it reads a MIB variable to determine the administrative state defined for each analog line.

Temporary Administrative State

You can set the administrative state of a line that will be kept until the Mediatrix 4100 restarts. Once the unit restarts, it uses the permanent state defined for each line. See [“Permanent Administrative State” on page 224](#) for more details.



In the *Unit Manager Network Administration Manual*, refer to chapter *Ports Parameters*, section *Port Configuration Window*.

You can also set these parameters via the web interface, as described in [“Interface Management” on page 34](#).

▶ To set a temporary administrative state:

1. In the *ifAdminMIB*, locate the *ifAdminSetAdmin* variable.

This variable temporarily locks/unlocks the selected line of the Mediatrix 4100. This state is kept until the unit restarts. It offers the following settings:

Table 135: Temporary Lock Settings

Setting	Description
unlock	Registers the line to the SIP server.
lock	Cancels the line registration to the SIP server. Active calls in progress remain established until normal call termination. No new calls may be initiated.
forcelock	Cancels the line registration to the SIP server. All active calls in progress are terminated immediately. No new calls may be initiated.

Permanent Administrative State

The permanent administrative state is applied every time the Mediatrix 4100 restarts.

► **To set a permanent administrative state:**

1. In the *ifAdminMIB*, locate the *ifAdminInitialAdminState* variable.
This variable indicates the administrative state the current analog line will have after the Mediatrix 4100 restarts. It offers the following settings:

Table 136: Permanent Lock Settings

Setting	Description
unlocked	Registers the line to the SIP server.
locked	The analog line is unavailable for normal operation. It cannot be used to make and/or receive calls.

Unregistered Line Behaviour

You can specify whether a line should remain enabled or not when not registered. This is useful if you want your users to be able to make calls even if the line is not registered with the SIP server.

This feature is currently located under the *mediatrixExperimental* branch of the MIB structure. See [“MIB Structure” on page 135](#) for more details.

► **To specify unregistered line behaviour:**

1. In the *sipExperimentalMIB*, locate the *sipUnregisteredPortBehavior* variable.
The following values are available:

Table 137: Unregistered Line Behaviour

Value	Description
disablePort	When the line is not registered, it is disabled. The user cannot make or receive calls. Picking up the handset yields a fast busy tone, and incoming INVITES receive a “403 Forbidden” response.
enablePort	When the line is not registered, it is still enabled. The user can receive and initiate outgoing calls. Note that because the line is not registered to a registrar, its public address is not available to the outside world; the line will most likely be unreachable except through direct IP calling.

Flash Hook Detection

The flash hook can be described as quickly depressing and releasing the plunger in or the actual handset-cradle to create a signal indicating a change in the current telephone session. Services such as picking up a call waiting, second call, call on hold, and conference are triggered by the use of the flash hook.

A flash hook is detected when the hook switch is pressed for a shorter time than would be required to be interpreted as a hang-up.

Using the “flash” button that is present on many standard telephone handsets can also trigger a flash hook.

The Mediatrix 4100 allows you to set the minimum and maximum time within which pressing and releasing the plunger is actually considered a flash hook.

► To set flash hook parameters:

1. In the *fxsMIB*, set the following variables:

Table 138: Flash Hook Parameters

Variable	Description
<i>fxsFlashHookDetectionDelayMin</i>	Minimum time in ms the hook switch must remain pressed to perform a flash hook. Default Value: 100
<i>fxsFlashHookDetectionDelayMax</i>	Maximum time in ms the hook switch can remain pressed to perform a flash hook. Default Value: 1200

2. Restart the Mediatix 4100 so that the changes may take effect.

Source Line Selection

The source line selection feature defines a list of callers that have the right to use a specific FXS line to make a call. This feature can be used to map an FXS line to a specific FXO line of a gateway such as the Mediatix 1204. See [“Examples of Source Line Selection Use” on page 225](#) for more details.

► To set the line selection:

1. In the *lineSelectionMIB*, define the list of telephone numbers that can use this line to make calls in the *lineSelectionDigitMap* variable.
Call sources that match this digit map can use this line. This string must follow the syntax for digit maps (see [“Chapter 21 - Digit Maps” on page 293](#)). This digit map will not have any effect unless the feature’s status is “enabled”.
Because this variable is located in a table, you can define different digit maps for each line of the Mediatix 4100.
2. Enable the line selection feature by setting the *lineSelectionEnable* variable to **enable**.
The source of the call is compared to all the source line selection digit maps defined in the previous step. The result of this comparison is a list of lines that can take the call, but are not necessarily available to do so.
Because this variable is located in a table, you can enable/disable the feature on a per-line basis.

Examples of Source Line Selection Use

FXS to FXO Line Mapping

You can map an FXS line to a specific FXO line of a gateway such as the Mediatix 1204. In this case, a call made from this FXS line will always use the same FXO line. To achieve that, the Mediatix 4100 and Mediatix 1204 configurations would be something similar to the following:

```
Mediatix 4100: IP address 192.168.0.1
sipUAMainUsername = 6661111 (FXS line #1)
sipUAMainUsername = 6662222 (FXS line #2)
sipUAMainUsername = 6663333 (FXS line #3)
sipUAMainUsername = 6664444 (FXS line #4)
```

```
Mediatix 1204: IP address 192.168.0.2
lineSelectionDigitMap (FXO line #1) = 6661111
telephonyAttributesAutomaticCallEnable = enable
telephonyAttributesAutomaticCallTargetAddress = 6661111
lineSelectionDigitMap (FXO line #2) = 6662222
```

```

telephonyAttributesAutomaticCallEnable = enable
telephonyAttributesAutomaticCallTargetAddress = 6662222
lineSelectionDigitMap (FXO line #3) = 6663333
telephonyAttributesAutomaticCallEnable = enable
telephonyAttributesAutomaticCallTargetAddress = 6662222
lineSelectionDigitMap (FXO line #4) = 6664444
telephonyAttributesAutomaticCallEnable = enable
telephonyAttributesAutomaticCallTargetAddress = 6662222

```

With such a configuration, a call made from line #2 of a Mediatrix 4100 is processed on line #2 of the Mediatrix 1204. On the other hand, if a caller from the SCN calls line #3 of the Mediatrix 1204, the call is automatically redirected to line #3 of the Mediatrix 4100.

Reserving an FXS Line

You can reserve an FXS line for specific individuals. For instance, these individuals could be the management team members of a company.

If the telephone numbers of the management team are 221 and 222 and you want to reserve an FXS line for their exclusive use, configure the Mediatrix 4100 as follows:

```

lineSelectionDigitMap (FXS line #1) = (221|222)
lineSelectionDigitMap (FXS line #2) = xxx
lineSelectionDigitMap (FXS line #3) = xxx
lineSelectionDigitMap (FXS line #4) = xxx

```

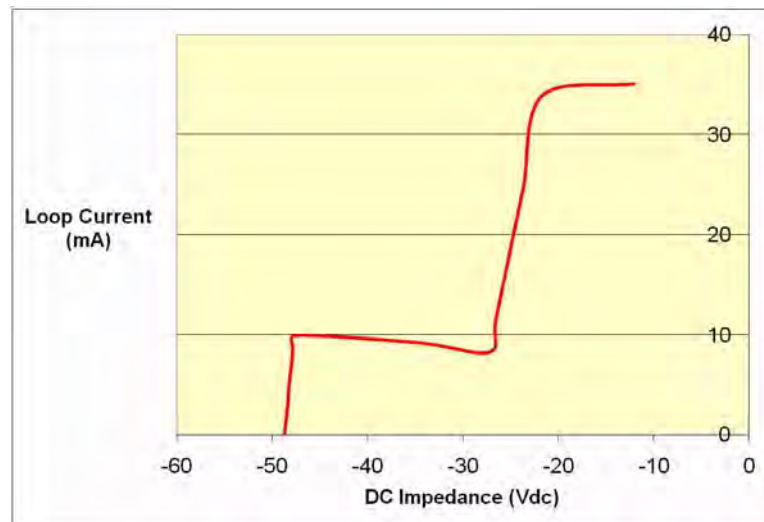
The management team can thus use all FXS lines, while others can only use lines 2,3 and 4.

Loop Current

When one of its analog lines goes off-hook, the Mediatrix 4100 controls the line in a fixed loop current mode. The value of the loop current can be modified through the MIB.

Note that the actual measured current may be different than the value you set, because it varies depending on the DC impedance. This is illustrated in [Figure 81](#) for a loop current of 32 mA.

Figure 81: Loop Current vs Impedance – 32 mA



► To set the loop current:

1. In the *fxsMIB*, set the *fxsLoopCurrent* variable to the value you want to use. The loop current is in mA. The range of available values is from 20 mA to 32 mA.

2. If applicable, configure the Mediatrix 4100 to suppress loop current on its lines when they cannot be used or when the IP connection is lost by setting the *fxsLoopCurrentDropEnable* variable to **enable**.

The loop current is interrupted on the port as soon as it enters an unusable state, or if the unit's IP network connection is lost.

If you set the value to **disable**, the loop current is unaffected by the port usability.

3. Restart the Mediatrix 4100 so that the changes may take effect.

When a remote end-user goes on-hook, the Mediatrix 4100 signals the far end disconnect by performing a current loop drop (< 1 mA) on the analog line. This current loop drop, also referred to as "Power Denial" mode, is typically used for disconnect supervision on analog lines. The Mediatrix 4100 maintains a current drop for one second (this value cannot be configured), then a busy tone is generated to indicate the user to hang up.

Callee Hang-up Supervision

This feature determines whether call clearing occurs as soon as the called user is the first to hang up a received call or after a user-defined delay.

This feature allows to emulate the behaviour of some SCNs that delay the ending of a call when the callee hangs up first.

When the feature is activated, hanging up on a received call will not terminate the call right away. Instead, the connection will remain active for a user-defined amount of time. If the callee picks up the phone before the expiration of that delay, he is still in communication with the caller (assuming the caller has not already hung up).



Note: This feature is effective only when the user is the called party. When the user acts as the calling party, the call ends as soon as the user hangs up. It is also effective only with a single active call. Callee hang-up supervision has no effect when the callee hangs up with a call on hold, wants to perform a call transfer, or if he is the initiator of a three-way conference call.

► To enable the callee hang-up supervision:

1. In the *fxsMIB*, set the *fxsCalleeHangupDelay* variable with the amount of time, in seconds, the Mediatrix 4100 waits after the called user hangs up before signalling the end of the call.
The default value is **60** seconds.
2. Enable the callee hang-up supervision feature by setting the *fxsCalleeHangupSupervision* variable to **enable**.

When the user is the first to hang up on a received call, the Mediatrix 4100 waits for the amount of time set in the *fxsCalleeHangupDelay* variable before signalling the end of the call.

Line Reversal

Two options are available to determine how the line polarity is used to signal the beginning and end of a call. They are used because of an inability by some customer's CPE to react to busy tone. When one of the options is activated, it replaces the default behaviour of the Mediatrix 4100, which is to briefly remove power from the line to signal that the remote party has hung up.

► **To enable line reversal:**

1. In the *fxsMIB*, set the *fxsPolarityAndDenialBehavior* variable with the proper parameter.

Table 139: Line Reversal Parameters

Parameter	Description
noReversal	Polarity reversal is not used. This is the default value. A short power drop is made at the end of a call when the call is disconnected by the remote party. The drop duration can be configured in the variable <i>fxsPowerDropOnDisconnectDuration</i> (Step 2).
reversalOnIdle	Activates the Reversal on Idle feature. When a line is in the idle, line lockout, or blocked state, the polarity of the line is reversed. When the line is seized to originate a call, or the ring current is applied to terminate a call, the line polarity is returned to normal. On release, the line polarity is returned to the reversed condition.
reversalOnEstablished	Activates the Reversal on Established option. The polarity remains normal (voltage on the phone line stays positive) when you pick up the phone, dial, and when the device is ringing. As soon as the other party picks up the phone, the unit reverses the line polarity. When the unit receives a hang-up signal or the caller hangs up the phone, the voltage returns to its positive state.

2. Set the *fxsPowerDropOnDisconnectDuration* variable with the power drop duration, in milliseconds, that is made at the end of a call when the call is disconnected by the remote party.
This variable only has an effect when the variable *fxsPolarityAndDenialBehavior* is set to **noReversal**.

Blanking of an Anonymous Caller ID

You can instruct the Mediatrix 4100 to blank out the name portion of the received caller ID whenever "anonymous" (case insensitive) is used to identify the originator of the call before the caller ID data is sent to the telephone. If the variable is not activated, the anonymous string is passed on to the telephone and displayed to identify the calling party.

► **To enable anonymous caller ID blanking:**

1. In the *fxsMIB*, set the *fxsBlankAnonymousCallerId* variable to **enable**.

Calling Number Transformation

It is possible to modify the value of the Caller ID before it is sent on the line. The Caller ID must match the criteria specified in the *fxsCallingNumberCriteria* variable (which must contain a regular expression). The transformation specified in the *fxsCallingNumberTransformation* variable is then applied.

► To modify the value of the Caller ID:

1. In the *fxsMIB*, set the *fxsCallingNumberCriteria* variable with the expression that the calling number must match in order to apply the transformation.
The criteria must be a regular expression. See [“Regular Expressions” on page 229](#) for more details. If you leave this variable empty, the feature is disabled.
2. Set the *fxsCallingNumberTransformation* variable with the transformation to apply on the calling number if it matches the expression specified in *fxsCallingNumberCriteria*.
You can use the “\0” to “\9” groups in the string. They are replaced by the corresponding group in the regular expression specified in *fxsCallingNumberCriteria*. See [“Groups” on page 230](#) for more details.

Regular Expressions

Standards Supported

- IEEE Std 1003.1-2001: IEEE Standard for Information Technology---Portable Operating System Interface (POSIX®)

A regular expression is a string used to find and replace strings in other large strings. The Mediatrix 4100 uses regular expressions to enter a value in the *fxsCallingNumberCriteria* variable, often by using wildcard characters. These characters provide additional flexibility and decrease the need for multiple entries in configuring number ranges.

The expression cannot begin by “^”, it is implicit in the expression. The following table shows some of the wildcard characters that are supported:

Table 140: Regular Expressions Wildcards

Character	Description
.	Single-digit place holder. For instance, 555 matches any dialed number beginning with 555, plus at least four additional digits. Note that the number may be longer and still match.
*	Repeats the previous digit 0, 1, or more times. For instance, in the pattern: 1888*1 the pattern matches: 1881, 18881, 188881, 1888881 Note: If you are trying to handle the asterisk (*) as part of a dialed number, you must use *.
[]	Range of digits. • A consecutive range is indicated with a hyphen (-), for instance, [5-7]. • A nonconsecutive range is indicated without a delimiter, for instance, [58]. • Both can be used in combination, for instance [5-79], which is the same as [5679]. You may place a (^) symbol right after the opening bracket to indicate that the specified range is an exclude list. For instance, [^01] specifies the same range as [2-9]. Note: The Mediatrix 4100 only supports single-digit ranges. You cannot specify the range of numbers between 99 and 102 by using [99-102].
()	Indicates a pattern (also called group), for instance, 555(2525). It is used when replacing a number in a mapping. See “Groups” on page 230 for more details.

Table 140: Regular Expressions Wildcards (Continued)

Character	Description
?	Matches 0 or 1 occurrence of the previous item. For instance, 123?4 matches both 124 and 1234.
+	Repeats the previous digit one or more time. For instance 12+345 matches 12345, 122345, etc. (but not 1345). If you use the + at the end of a number, it repeats the last number one or more times. For instance: 12345+ matches, 12345, 123455, 1234555, etc.

The matching criterion implicitly matches from the beginning of the string, but not necessarily up to the end. For instance, 123 will match the criterion 1, but it will not match the criterion 2.

If you want to match the whole string, you must end the criterion with "\$". For instance, 123 will not match the criterion 1\$ and will match the criterion 123\$.

Groups

A group is placed within parenthesis. It is used when replacing a string in a mapping. You can use up to nine groups (defined by "\1" to "\9") and matching is not case sensitive. "\0" represents the whole string. Lets say for instance you have the following string:

9(123(45)6)

The following describes how the groups are replaced:

Table 141: Groups Replacement Example

Replacement	Result
\0	9123456
\1	123456
\2	45
\3	

This chapter describes the various codecs the Mediatrix 4100 supports for transmitting audio signals.

You can also set some of these parameters via the web interface, as described in [“Voice & Fax Codecs” on page 79](#).

Codec Descriptions

The lines of the Mediatrix 4100 can simultaneously use the same codec (for instance, G.711 PCMA), or a mix of any of the supported codecs. Set and enable these codecs for **each** line.

Table 142: Codecs Comparison

	Compression	Voice Quality
G.711	None	Excellent
G.723.1	Highest	Good
G.726	Medium	Fair
G.729a/ab	High	Fair/Good

G.711 PCMA and PCMU

Specified in ITU-T Recommendation G.711. The audio data is encoded as 8 bits per sample, after logarithmic scaling. PCMU denotes μ -law scaling, PCMA A-law scaling.

Table 143: G.711 Features

Feature	Description
Packetization time	Range of 10 ms to 100 ms with increment of 10 ms. See “Packetization Time” on page 234 for more details.
Voice Activity Detection (VAD)	Can be enabled or disabled. When enabled, two levels of detection are available: transparent or conservative. See “G.711 and G.726 VAD” on page 244 for more details.
Comfort noise	Supports white and custom comfort noise as defined in <i>RFC 3389</i> . See “Comfort Noise” on page 246 for more details.

Analog Modem

The Mediatrix 4100 can send modem transmissions in clear channel (G.711). If configured adequately, modems with higher rate capabilities (for instance, V.90) will automatically fall back in the transmission range supported.

Quality of modem transmissions is dependent upon the system configuration, quality of the analog lines, as well as the number of analog-to-digital and digital-to-analog conversions. Modem performance may therefore be reduced below the optimum values stated above.

G.726

Specified in ITU-T Recommendation G.726: 40, 32, 24, 16 kbit/s adaptive differential pulse code modulation (ADPCM). It describes the algorithm recommended for conversion of a single 64 kbit/s A-law or U-law PCM channel encoded at 8000 samples/sec to and from a 40, 32, 24, or 16 kbit/s channel. The conversion is applied to the PCM stream using an Adaptive Differential Pulse Code Modulation (ADPCM) transcoding technique.

Table 144: G.726 Features

Feature	Description
Packetization time	Range of 10 ms to 100 ms with increment of 10 ms. The preferred is 20 ms. See "Packetization Time" on page 234 for more details.
Voice Activity Detection (VAD)	Uses the G.711 VAD settings. Can be enabled or disabled. When enabled, two levels of detection are available: transparent or conservative. See "G.711 and G.726 VAD" on page 244 for more details.
Comfort noise	Uses the G.711 comfort noise settings. Supports white and custom comfort noise as defined in RFC 3389. See "Comfort Noise" on page 246 for more details.

Analog Modem

The Mediatrix 4100 can send modem transmissions in clear channel (G.726). If configured adequately, modems with higher rate capabilities (for instance, V.90) will automatically fall back in the transmission range supported

Quality of modem transmissions is dependent upon the system configuration, quality of the analog lines, as well as the number of analog-to-digital and digital-to-analog conversions. Modem performance may therefore be reduced below the optimum values stated above.

G.723.1

Specified in ITU-T Recommendation G.723.1, dual-rate speech coder for multimedia communications transmitting at 5.3 kbit/s and 6.3 kbit/s. This Recommendation specifies a coded representation that can be used to compress the speech signal component of multi-media services at a very low bit rate. The audio is encoded in 30 ms frames.

A G.723.1 frame can be one of three sizes: 24 octets (6.3 kb/s frame), 20 octets (5.3 kb/s frame), or 4 octets. These 4-octet frames are called SID frames (Silence Insertion Descriptor) and are used to specify comfort noise parameters.

Table 145: G.723.1 Features

Feature	Description
Packetization time	Range of 30 ms to 120 ms with increment of 30 ms. See "Packetization Time" on page 234 for more details.
Voice Activity Detection (VAD)	The Mediatrix 4100 supports the annex A. Annex A is the built-in support of VAD in G.723.1.

G.729

Specified in ITU-T Recommendation G.729, coding of speech at 8 kbit/s using conjugate structure-algebraic code excited linear prediction (CS-ACELP). For all data rates, the sampling frequency (and RTP timestamp clock rate) is 8000 Hz.

A voice activity detector (VAD) and comfort noise generator (CNG) algorithm in Annex B of G.729 is recommended for digital simultaneous voice and data applications; they can be used in conjunction with G.729 or G.729 Annex A. A G.729 or G.729 Annex A frame contains 10 octets, while the G.729 Annex B comfort noise frame occupies 2 octets.

The Mediatrix 4100 supports G.729A and G.729AB for encoding and G.729, G.729A and G.729AB for decoding.

Table 146: G.729 Features

Feature	Description
Packetization time	Range of 10 ms to 100 ms with increment of 10 ms. See “Packetization Time” on page 234 for more details.
Voice Activity Detection (VAD)	The Mediatrix 4100 supports the annex B. Annex B is the built-in support of VAD in G.729. See “G.729 VAD” on page 244 for more details.

Preferred Codec

The preferred codec is the codec you want to favour during negotiation.



In the *Unit Manager Network Administration Manual*, refer to chapter *Ports Parameters*, section *Port Configuration Window*.

► To set a preferred codec:

1. In the *voicelfMIB*, locate the *voicelfCodecPreferred* variable (*voicelfCodecTable*). This variable sets the preferred codec for this line.
2. Choose the codec you want to use from one of the available configurations:
 - pcmu
 - pcma
 - g723
 - g729
 - g726-16kbps
 - g726-24kbps
 - g726-32kbps
 - g726-40kbps

The default value is **pcmu**.

Enabling Individual Codecs

Enabling individual codecs allows you to define codecs that can be considered during negotiation. If codecs are disabled, they are not considered.

► To enable voice codecs:

1. In the *voicelfMIB*, choose the codec you want to use (*voicelfCodecTable*). You have the choice between the following codecs:

Table 147: Enabling Voice Codecs

Codec	Variable	Set to...
PCMU (G.711 u-Law)	voicelfCodecPcmuEnable	enable
PCMA (G.711 a-Law)	voicelfCodecPcmaEnable	enable

Table 147: Enabling Voice Codecs (Continued)

Codec	Variable	Set to...
G.723.1	voicelfCodecG723Enable	g723-53kbs g723-63kbs
G.726 at 16 kbps	voicelfCodecG72616kbpsEnable	enable
G.726 at 24 kbps	voicelfCodecG72624kbpsEnable	enable
G.726 at 32 kbps	voicelfCodecG72632kbpsEnable	enable
G.726 at 40 kbps	voicelfCodecG72640kbpsEnable	enable
G.729.A	voicelfCodecG729Enable	enable

2. If you have enabled one or more of the G.726 codecs, set the G.726 actual RTP dynamic payload type used in an initial offer in one or more of the following variables:
- `voicelfCodecG72616kbpsPayloadType`: The default value is 97.
 - `voicelfCodecG72624kbpsPayloadType`: The default value is 98.
 - `voicelfCodecG72632kbpsPayloadType`: The default value is 99.
 - `voicelfCodecG72640kbpsPayloadType`: The default value is 100.

The payload types available are as per RFC 3551. The values range from 96 to 127.



Note: When selecting the dynamic payload type, make sure that the value is not already used by another dynamic codec. If a value between 96 and 127 is refused, this means it is already used by another dynamic codec.



Note: If you set the `voicelfDtmfTransport` variable to **outOfBandUsingSignalingProtocol** ([“DTMF Transport Type” on page 237](#)), you cannot configure a dynamic payload type to 111 because it is already used by the DTMF out-of-band using signalling protocol.

3. Restart the Mediatrix 4100 so that the changes may take effect.

Packetization Time

The packetization time (also called packetization period or `ptime`) is the duration, in ms, of the voice packet.

► To set the packetization time:

1. In the `voicelfMIB`, set the packetization time of the codec(s) as required (`voicelfCodecTable`). Available values vary from one codec to another.

Table 148: Packetization Time Settings

Variable	Definition	Values (ms)
PCMU (G.711 u-Law)		
<code>voicelfCodecPcmuMinPTime</code>	Shortest packetization period allowed for the PCMU codec. Authorized values start at 10 ms and come in discrete steps of 10 ms up to the one specified by the <code>voicelfCodecPcmuMaxPTime</code> variable. Default Value: 10	10-100, with increments of 10

Table 148: Packetization Time Settings (Continued)

Variable	Definition	Values (ms)
voicelfCodecPcmuMaxPTime	Longest packetization period allowed for the PCMU codec. Authorized values go up to 100 ms, in discrete steps of 10 ms, and start at the one specified by the <i>voicelfCodecPcmuMinPTime</i> variable. Default Value: 100	10-100, with increments of 10
PCMA (G.711 a-Law)		
voicelfCodecPcmaMinPTime	Shortest packetization period allowed for the PCMA codec. Authorized values start at 10 ms and come in discrete steps of 10 ms up to the one specified by the <i>voicelfCodecPcmaMaxPTime</i> variable. Default Value: 10	10-100, with increments of 10
voicelfCodecPcmaMaxPTime	Longest packetization period allowed for the PCMA codec. Authorized values go up to 100 ms, in discrete steps of 10 ms, and start at the one specified by the <i>voicelfCodecPcmaMinPTime</i> variable. Default Value: 100	10-100, with increments of 10
G.723		
voicelfCodecG723MinPTime	Shortest packetization period allowed for the G.723 codec. Authorized values start at 30 ms and come in discrete steps of 30 ms up to the one specified by the <i>voicelfCodecG723MaxPTime</i> variable. Default Value: 30	30, 60, 90, 120
voicelfCodecG723MaxPTime	Longest packetization period allowed for the G.723 codec. Authorized values go up to 120 ms, in discrete steps of 30 ms, and start at the one specified by the <i>voicelfCodecG723MinPTime</i> variable. Default Value: 120	30, 60, 90, 120
G.726		
voicelfCodecG72616kbpsMinPTime	Shortest packetization period allowed for the G.726-16kbps codec. Authorized values start at 10 ms and come in discrete steps of 10 ms up to the one specified by the <i>voicelfCodecG72616kbpsMaxPTime</i> variable. Default Value: 10	10-100, with increments of 10
voicelfCodecG72616kbpsMaxPTime	Longest packetization period allowed for the G.726-16kbps codec. Authorized values go up to 100 ms, in discrete steps of 10 ms, and start at the one specified by the <i>voicelfCodecG72616kbpsMinPTime</i> variable. Default Value: 100	10-100, with increments of 10

Table 148: Packetization Time Settings (Continued)

Variable	Definition	Values (ms)
voicelfCodecG72624kbpsMinPTime	Shortest packetization period allowed for the G.726-24kbps codec. Authorized values start at 10 ms and come in discrete steps of 10 ms up to the one specified by the <i>voicelfCodecG72624kbpsMaxPTime</i> variable. Default Value: 10	10-100, with increments of 10
voicelfCodecG72624kbpsMaxPTime	Longest packetization period allowed for the G.726-24kbps codec. Authorized values go up to 100 ms, in discrete steps of 10 ms, and start at the one specified by the <i>voicelfCodecG72624kbpsMinPTime</i> variable. Default Value: 100	10-100, with increments of 10
voicelfCodecG72632kbpsMinPTime	Shortest packetization period allowed for the G.726-32kbps codec. Authorized values start at 10 ms and come in discrete steps of 10 ms up to the one specified by the <i>voicelfCodecG72632kbpsMaxPTime</i> variable. Default Value: 10	10-100, with increments of 10
voicelfCodecG72632kbpsMaxPTime	Longest packetization period allowed for the G.726-32kbps codec. Authorized values go up to 100 ms, in discrete steps of 10 ms, and start at the one specified by the <i>voicelfCodecG72632kbpsMinPTime</i> variable. Default Value: 100	10-100, with increments of 10
voicelfCodecG72640kbpsMinPTime	Shortest packetization period allowed for the G.726-40kbps codec. Authorized values start at 10 ms and come in discrete steps of 10 ms up to the one specified by the <i>voicelfCodecG72640kbpsMaxPTime</i> variable. Default Value: 10	10-100, with increments of 10
voicelfCodecG72640kbpsMaxPTime	Longest packetization period allowed for the G.726-40kbps codec. Authorized values go up to 100 ms, in discrete steps of 10 ms, and start at the one specified by the <i>voicelfCodecG72640kbpsMinPTime</i> variable. Default Value: 100	10-100, with increments of 10
G.729		
voicelfCodecG729MinPTime	Shortest packetization period allowed for the G.729 codec. Authorized values start at 10 ms and come in discrete steps of 10 ms up to the one specified by the <i>voicelfCodecG729MaxPTime</i> variable. Default Value: 10	10-100, with increments of 10
voicelfCodecG729MaxPTime	Longest packetization period allowed for the G.729 codec. Authorized values go up to 100 ms, in discrete steps of 10 ms, and start at the one specified by the <i>voicelfCodecG729MinPTime</i> variable. Default Value: 100	10-100, with increments of 10



Note: The packetization time is not negotiated between endpoints, so a minimum and a maximum don't make much sense. The selected value is the default RTP value (20 ms for G.711, G.726, and G.729.AB) if it is included in the range delimited by the minimum and maximum. Otherwise, it is the minimum.

DTMF Transport Type

Standards Supported

- draft-choudhuri-sip-info-digit-00.txt
- ITU-T Recommendation Q.24 : Multifrequency push-button signal reception
- RFC 2833: RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals

You can define how to transport the DTMFs.

► To set the DTMF transport type:

1. In the *voicelfMIB*, set the DTMF transport type in the *voicelfDtmfTransport* variable (*voicelfDtmfTransportTable* group).

The following choices are available:

Table 149: DTMF Transport Type Parameters

Transport Parameter	Description
inBand	The DTMFs are transmitted like the voice in the RTP stream.
outOfBandUsingRtp	The DTMFs are transmitted as per RFC 2833 (see “DTMF Payload Type” on page 238 for additional information).
outOfBandUsingSignalingProtocol	The DTMFs are transmitted as per <i>draft-choudhuri-sip-info-digit-00.txt</i> (see “DTMF Transport Using SIP INFO” on page 238 and “DTMF Transport over the SIP Protocol” on page 239 for more details). Note: This feature and the Hook Flash processing feature via signalling protocol are totally independent. Activating one of these features has no effect on the other. See “Hook Flash Processing” on page 324 for more details.
signalingProtocolDependent	The signalling protocol has the control to select the DTMF transport mode. The SDP body includes both RFC 2833 and <i>draft-choudhuri-sip-info-digit-00.txt</i> in that order of preference.

DTMF out-of-band

Certain compression codecs such as G.723.1 and G.729 effectively distort voice because they lose information from the incoming voice stream during the compression and decompression phases. For normal speech this is insignificant and becomes unimportant. In the case of pure tones (such as DTMF) this distortion means the receiver may no longer recognize the tones. The solution is to send this information as a separate packet to the other endpoint, which then plays the DTMF sequence back by re-generating the true tones. Such a mechanism is known as out-of-band DTMF. The Mediatrix 4100 receives and sends out-of-band DTMFs as per ITU Q.24. DTMFs supported are 0-9, A-D, *, #.

DTMF Transport Using SIP INFO

Standards Supported

- RFC 2976: The SIP INFO Method
- draft-choudhuri-sip-info-digit-00.txt

You can use the SIP INFO method to collect and transport DTMFs. The collection process is regarded as being an unsolicited one-character timer-less digit collection.

When the feature is enabled:

- ▶ The Mediatrix 4100 sends a separate SIP INFO method every time a digit is entered during the call.
- ▶ The Mediatrix 4100 plays each DTMF sent in a separate message upon receiving a valid SIP INFO message.

▶ To enable the DTMF transport using the SIP INFO feature:

1. In the *voicelfMIB*, set the DTMF transport type in the *voicelfDtmfTransport* variable (*voicelfDtmfTransportTable* group) according to the transport type you want to use.

There are three methods to transport DTMF events:

- in-band
- out-of-band using RTP (RFC 2833)
- out-of-band using SIP INFO

Table 150: Transport Type Setting

To	Set the variable to:
Transport DTMF events in-band.	inBand
Transport DTMF events out-of-band by exclusively using RTP (RFC 2833).	outOfBandUsingRtp
Transport DTMF events out-of-band by exclusively using the SIP INFO method.	outOfBandUsingSignalingProtocol
Offer the choice to transport DTMF events out-of-band by using either RTP or the SIP INFO method.	signalingProtocolDependent

If you set the *voicelfDtmfTransport* variable to **signalingProtocolDependent**, the remote party must select one of the two transport types. Transporting DTMF by using RTP has priority over the SIP INFO method.

DTMF Payload Type

Standards Supported

- RFC 1890 – RTP Profile for Audio and Video Conferences with Minimal Control

When selecting the *outOfBandUsingRtp* DTMF transport mode (see [“DTMF Transport Type” on page 237](#) for more details), you can determine the actual RTP dynamic payload type used for the “telephone-event” in an initial offer. The payload types available are as per RFC 1890.

▶ To define the DTMF payload type:

1. In the *voicelfMIB*, set the DTMF transport type in the *voicelfDtmfTransport* variable (*voicelfDtmfTransportTable* group) to **outOfBandUsingRtp**.
2. Set the payload type in the *voicelfDtmfPayloadType* variable.
Available values range from 96 to 127.

DTMF – RFC 2833 Events

Standards Supported

- RFC 2833: RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals

You can define which events will be relayed via RFC 2833. This could be very useful in a Remote Line Extension scenario, as described in [“Remote Line Extension” on page 327](#).

► To define the DTMF enforce default events:

1. In the *voicelfMIB*, set the DTMF enforce default events in the *voicelfDtmfEnforceDefaultEvents* variable.

Table 151: DTMF Enforce Default Events

Parameter	Description
enable	Conformance is enforced and support for RFC 2833 implies the support of basic telephony-events. When setting the variable <i>voicelfDtmfTransport</i> to outOfBandUsingRtp (“DTMF Transport Type” on page 237), or the variable <i>telephonyAttributesHookFlashProcessing</i> to outOfBandUsingRtp (“Hook Flash Processing” on page 324), the Mediatrix 4100 will advertise the support for events 0-15; it will assume support for events 0-15 when support for RFC 2833 is received in an announcement.
disable	This creates a deliberate deviance to RFC 2833 as support of basic events is not automatic. The variables <i>voicelfDtmfTransport</i> and <i>telephonyAttributesHookFlashProcessing</i> then act independently to specify which events will be relayed via RFC 2833. If Hook Flash relay is enabled by itself, support of event 16 alone will be advertised; if both Hook Flash and DTMF relay are activated, events 0-16 are supported.

DTMF Transport over the SIP Protocol

Standards Supported

- draft-choudhuri-sip-info-digit-00.txt

You can select the method used to transport DTMFs out-of-band over the SIP protocol.

This feature is effective only if the *voicelfDtmfTransport* variable is set to **outOfBandUsingSignalingProtocol** (see [“DTMF Transport Type” on page 237](#) for more details).

► To select the DTMF transport method over the SIP protocol:

1. In the *voicelfMIB*, set the DTMF transport type in the *voicelfDtmfTransport* variable to **outOfBandUsingSignalingProtocol**.
2. In the *sipInteropMIB*, set the DTMF transport type in the *sipInteropDtmfTransportMethod* variable (*sipInteropDtmfTransportBySipProtocol* group).

The following methods are available:

Table 152: DTMF Out-of-Band Transport Methods

Method	Description
draftChoudhuriSipInfoDigit00	Transmits DTMFs by using the method defined in <i>draft-choudhuri-sip-info-digit-00</i> . Only the unsolicited-digit part is supported.

Table 152: DTMF Out-of-Band Transport Methods (Continued)

Method	Description
infoDtmfRelay	Transmits DTMFs by using a custom method. This custom method requires no SDP negotiation and assumes that the other peer uses the same method. It uses a SIP INFO message with a content of type <i>application/dtmf-relay</i>. The body of the message contains the DTMF transmitted and the duration of the DTMF: <pre>Signal= 1 Duration= 160</pre> When transmitting, the duration is the one set in the <i>sipInteropDtmfTransportDuration</i> variable (see Step 3 below). When receiving, the duration of the DTMF received is not used. The value is the one set in the <i>analogScnGwDtmfDuration</i> variable (see Step 4 below). DTMFs are transmitted one at a time. It is thus not compatible with the PIN dialing feature (see “PIN Dialing” on page 326 for more details). Available digits are “0123456789ABCD*#”. The Mediatrix 4100 also supports the “;p” characters when receiving DTMFs.

- Set the DTMF duration sent in the INFO message when using the **infoDtmfRelay** method to transmit DTMFs in the *sipInteropDtmfTransportDuration* variable.
This value is expressed in milliseconds (ms). The default value is **100** ms.
- In the *analogScnGwMIB*, set the DTMF duration when using the **infoDtmfRelay** method to receive DTMFs in the *analogScnGwDtmfDuration* variable.
This is the duration, in milliseconds (ms), a DTMF is played when dialing the destination phone number.
- Set an inter-digit dial delay in the *analogScnGwInterDigitDial Delay* variable.
This is the delay, in milliseconds (ms), between two DTMFs when dialing the destination phone number. This is useful when the Mediatrix 4100 receives DTMFs out-of-band faster than it can signal them.
- Restart the Mediatrix 4100 so that the changes may take effect.

DTMF Detection

The default DTMF detection parameters of the Mediatrix 4100 may sometimes not be enough to properly detect the DTMFs. This section describes how to set additional DTMF detection parameters.

DTMF Frequencies

The DTMF keypad is laid out in a 4x4 matrix, with each row representing a low frequency, and each column representing a high frequency. For example, pressing a single key (such as '1') sends a sinusoidal tone of the two frequencies (697 Hz and 1209 Hz). When the unit is configured to send DTMFs out-of-band, its DSP detects these DTMFs, removes them from the RTP stream, and sends them out-of-band.

Table 153: DTMF Keypad Frequencies

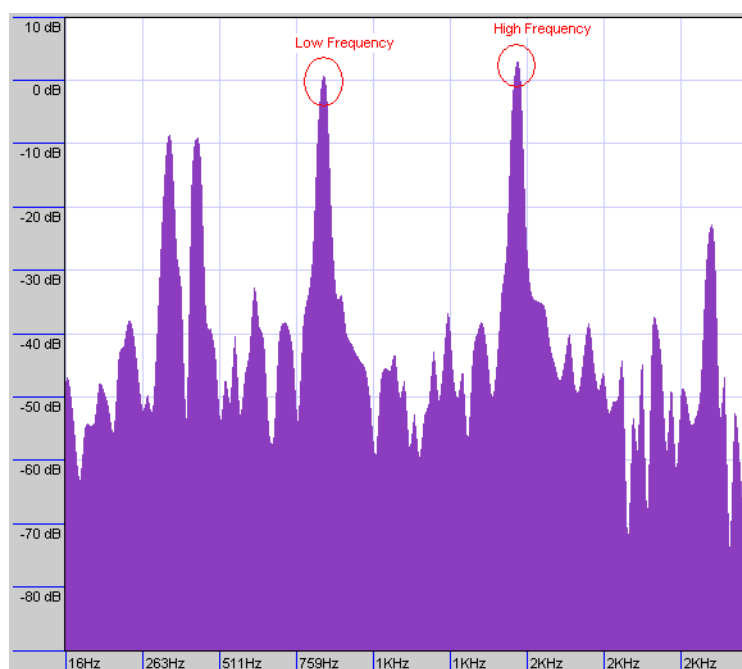
Low/High (Hz)	1209	1336	1477	1633
697	1	2	3	A
770	4	5	6	B

Table 153: DTMF Keypad Frequencies (Continued)

Low/High (Hz)	1209	1336	1477	1633
852	7	8	9	C
941	*	0	#	D

DTMF Detection Configuration

Below is a frequency spectrum analysis of a DTMF (9) with the Frequency in Hertz on the x axis and the Power in dBm on the y axis. The low and high frequencies of the DTMF are in red and you can clearly see that they are the most powerful frequencies in the signal.

Figure 82: DTMF Detection Example

To detect this DTMF, the DSP relies on several parameters. The following table lists the default values that the Mediatrix 4100 uses. You can override any one of these values.

Table 154: DTMF Detection Default Parameters

Parameter	Value
MinPowerThreshold	-30 dBm0
MaxPowerThreshold	1 dBm0
BreakPowerThreshold	32 dBm0
PositiveTwist	6 dBm0
NegativeTwist	9 dBm0
RiseTimeCriteria	confirmSnr

► To configure the DTMF detection:

1. In the *voiceIfMIB*, set the maximum absolute power threshold (dBm0) for the low and high frequencies in a DTMF in the *voiceIfDtmfDetectionUnitMaxPowerThreshold* variable.
The high AND low DTMF frequencies MUST be lower than this threshold otherwise the DTMF is not detected.

Raising this value increases the sensitivity of the DTMF detection. Raising this value too high may also cause false detections of DTMFs.

2. Set the minimum absolute power threshold (dBm0) for the low and high frequencies in a DTMF in the *voicelfDtmfDetectionUnitMinPowerThreshold* variable.

The high AND low DTMF frequencies MUST be higher than this threshold otherwise the DTMF is not detected.

You could, for instance use one of the following settings:

- -15 dBm0: This configuration detects even more false DTMFs in the voice pattern.
- -20 dBm0: This configuration detects more DTMFs in the voice pattern.
- -35 dBm0: his configuration detects less DTMFs in the voice pattern.
- -40 dBm0: This configuration detects even less DTMFs in the voice pattern.

Raising this value reduces the sensitivity to DTMF detection.

3. Set the break absolute power threshold (dBm0) for on-off transition of a DTMF in the *voicelfDtmfDetectionUnitBreakPowerThreshold* variable.

While a DTMF has been positively detected, the DTMF will be considered OFF as soon as the high OR low frequency in the DTMF gets below this threshold.

4. Set the *voicelfDtmfDetectionUnitPositiveTwist* variable.

When the high-group frequency of a DTMF is more powerful than the low-group frequency, the difference between the high-group frequency absolute power and the low-group frequency absolute power must be smaller than or equal to the value set in this variable. Otherwise, the DTMF is not detected.

Raising this value increases the sensitivity of DTMF detection. Raising this value too high may also cause false detections of DTMFs.

5. Set the *voicelfDtmfDetectionUnitNegativeTwist* variable.

When the low frequency of a DTMF is more powerful than the high frequency, the difference between the low frequency absolute power and the high frequency absolute power MUST be smaller than or equal to the value set in this variable. Otherwise, the DTMF is not detected.

Raising this value increases the sensitivity of DTMF detection. Raising this value too high may also cause false detections of DTMFs.

6. Define how the Rise Time criteria should be configured for DTMF detection in the *voicelfDtmfDetectionRiseTimeCriteria* variable.

Table 155: RiseTimeCriteria Parameters

Parameter	Description
checkSr	Enable the Step Rise criteria and disable the Confirm DTMF SNR criteria.
confirmSnr	Enable the Confirm DTMF SNR criteria and disable the Step Rise criteria.

The Step Rise criteria compares the current frame energy to the high frequency power of the previous frame. If the current frame energy is high enough, then it passes the test, further validating the DTMF.

Disabling the Step Rise criteria may result in deteriorated talk-off performance, but increases the detection of malformed DTMF.

The Confirm DTMF SNR criteria is an additional Signal-to-noise ratio test performed before a confirmed DTMF report is sent to finally validate the DTMF.

Adaptative Jitter Buffer

The jitter buffer allows better protection against packet loss, but increases the voice delay. If the network to which the Mediatrix 4100 is connected suffers from a high level of congestion, the jitter buffer protection level should be higher. If the network to which the Mediatrix 4100 is connected suffers from a low level of congestion, the jitter buffer protection level should be lower.



Note: Do not put 0 as values for the *voicelfTargetJitterBufferLength* and *voicelfMaxJitterBufferLength* variables.

► To set Jitter Buffer variables:

1. In the *voicelfMIB*, locate the *voicelfTable* group.
2. Define the jitter buffer length in the *voicelfTargetJitterBufferLength* variable.
 The adaptive jitter buffer attempts to hold packets to the target holding time. This is the minimum delay the jitter buffer adds to the system. The target jitter buffer length is in ms and must be equal to or smaller than the maximum jitter buffer.
 Values range from 0 ms to 135 ms. The default value is 30 ms. You can change values by increments of 1 ms, but Media5 recommends to use multiple of 5 ms.
 It is best not to set target jitter values below the default value. Setting a target jitter buffer below 5 ms could cause an error. Jitter buffer adaptation behaviour varies from one codec to another. See [“About Changing Jitter Buffer Values” on page 243](#) for more details.
3. Define the maximum jitter buffer length in the *voicelfMaxJitterBufferLength* variable.
 This is the maximum jitter the adaptive jitter buffer can handle. The jitter buffer length is in ms and must be equal to or greater than the target jitter buffer.
 Values range from 0 ms to 135 ms. The default value is 125 ms. You can change values by increments of 1 ms, but Media5 recommends to use multiple of 5 ms.
 The maximum jitter buffer value should be equal to the minimum jitter buffer value + 4 times the ptime value. Let's say for instance that:
 - Minimum jitter buffer value is 30 ms
 - Ptime value is 20 ms
 The maximum jitter buffer value should be: $30 + 4 \times 20 = 110$ ms
 See [“About Changing Jitter Buffer Values” on page 243](#) for more details.
4. Restart the Mediatrix 4100 so that the changes may take effect.

About Changing Jitter Buffer Values

Media5 recommends to avoid changing the target and maximum jitter buffer values unless experiencing or strongly expecting one of the following symptoms:

- If the voice is scattered, try to increase the maximum jitter buffer value.
- If the delay in the voice path (end to end) is too long, you can lower the target jitter value, but ONLY if the end-to-end delay measured matches the target jitter value.

For instance, if the target jitter value is 50 ms, the maximum jitter is 135 ms and the delay measured is 130 ms, it would serve nothing to reduce the target jitter. However, if the target jitter value is 100 ms and the measured delay is between 100 ms and 110 ms, then you can lower the target jitter from 100 ms to 30 ms.

Voice Activity Detection

The Voice Activity Detection (VAD) defines how the Mediatrix 4100 sends information pertaining to silence. This allows the unit to detect when the user talks, thus avoiding to send silent RTP packets. This saves on network resources. However, the VAD may affect packets that are not really silent (for instance, cut sounds that are too low). The VAD can thus slightly affect the voice quality.



Note: You cannot disable the G.723 VAD.

G.711 and G.726 VAD

The G.711/G.726 VAD is generic – when enabling VAD, G.711/G.726 sends speech frames only during periods of audio activity. During silence periods, G.711/G.726 does not send speech frames, but it may send Comfort Noise (CN) packets (payload 13) containing information about background noise.



In the *Unit Manager Network Administration Manual*, refer to chapter *Ports Parameters*, section *Port Configuration Window*.

► **To enable G.711 and G.726 VAD:**

1. In the *voicelfMIB*, locate the *voicelfTable* group.
2. Define the sensitivity of the VAD algorithm to silence periods in the *voicelfG711VoiceActivityDetectionEnable* variable.

The following settings are available:

Table 156: G.711/G.726 VAD Settings

Setting	Description
Disable	VAD is not used.
Transparent	VAD is enabled. It has low sensitivity to silence periods.
Conservative	VAD is enabled. It has normal sensitivity to silence periods.

The difference between transparent and conservative is how “aggressive” the algorithm considers something as an inactive voice and how “fast” it stops the voice stream. A setting of conservative is a little bit more aggressive to react to silence compared to a setting of transparent.

The default value is **conservative**.

3. Restart the Mediatrix 4100 so that the changes may take effect.

G.729 VAD

G.729 has a built-in VAD in its Annex B version. It is recommended for digital simultaneous voice and data applications and can be used in conjunction with G.729 or G.729 Annex A. A G.729 or G.729 Annex A frame contains 10 octets, while the G.729 Annex B frame occupies 2 octets. The CN packets are sent in accordance with annex B of G.729.

► **To enable G.729 VAD:**

1. In the *voicelfMIB*, locate the *voicelfTable* group.
2. Define the *voicelfG729VoiceActivityDetectionEnable* variable.

The following settings are available:

Table 157: G.729 VAD Settings

Setting	Description
disable	G.729 uses annex A only. The Mediatix 4100 does not send G.729 Annex B comfort noise frames.
enable	G.729 annex A is used with annex B. The Mediatix 4100 sends G.729 Annex B comfort noise frames during silence periods.

See [“Enabling Individual Codecs” on page 233](#) for more details.

Echo Cancellation

Echo cancellation eliminates the echo effect caused by signal reflections. An echo is a signal that has been reflected or otherwise returned with enough magnitude and delay to be perceived. The echo cancellation is usually an active process in which echo signals are measured and cancelled or eliminated by combining an inverted signal with the echo signal.

You can disable the echo cancellation feature of the Mediatix 4100, which may be useful to ensure the success of some modem transmissions.

► **To enable or disable echo cancellation:**

1. In the *voicelfMIB*, set the *voicelfEchoCancellationEnable* variable to the proper value.

Table 158: Echo Cancellation Parameters

Parameter	Description
disable	The DSP does not use echo cancellation on the related port.
enable	The DSP proceeds to cancel signals that are recognized as echo when appropriate. This is the default value.

Signal Limiter

If you are experiencing echo issues, you may use the Signal Limiter. When enabled, the Signal Limiter attenuates or removes peaks in the voice signal.



Caution: This may reduce the presence of echo but may impair DTMF detection because of the signal distortion.

► **To use the Signal Limiter:**

1. In the *voicelfMIB*, set the *voicelfSignalLimiterLevel* variable to the proper value.

Table 159: Signal Limiter Parameters

Parameter	Description
0	Disable signal limiter
1	Most aggressive
...	
5	Least aggressive

Comfort Noise

Standards Supported

- RFC 3389: Real-time Transport Protocol (RTP) Payload for Comfort Noise (CN)

Comfort Noise (CN) defines how the Mediatrix 4100 processes silence periods information it receives.



Note: Comfort noise only applies to the G.711 and G.726 codecs. G.723 and G.729 CNG is not configurable because it is part of the codec.

During silence periods, the Mediatrix 4100 may receive CN packets containing information about background noise. When enabling Comfort Noise Generation (CNG), those packets are used to generate local comfort noise.

► To enable Comfort Noise:

1. In the *voicelfMIB*, locate the *voicelfTable* group.
2. Define the type of comfort noise in the *voicelfG711ComfortNoiseGenerationEnable* variable. The following settings are available:

Table 160: Comfort Noise Settings

Setting	Description
disable	CNG disabled.
whiteNoise	CNG enabled – white noise.
customNoise	CNG enabled – custom noise. More elaborated background noise that sounds better than white comfort noise.

3. Restart the Mediatrix 4100 so that the changes may take effect.

User Gain

The user gain allows you to modify the input and output sound level of the Mediatrix 4100.



Caution: Use these settings with great care. Media5 recommends not to modify the user gain variables unless absolutely necessary because default calibrations may not be valid anymore.

Modifying user gains may cause problems with DTMF detection and voice quality – using a high user gain may cause sound saturation (the sound is distorted). Furthermore, some fax or modem tones may not be recognized anymore. The user gains directly affect the fax communication quality and may even prevent a fax to be sent.

You can compensate with the user gain if there is no available configuration for the country in which the Mediatrix 4100 is located. Because the user gain is in dB, you can easily adjust the loss plan (e.g., if you need an additional 1 dB for analog to digital, simply put 1 for user gain input).



In the *Unit Manager Network Administration Manual*, refer to chapter *Ports Parameters*, section *Port Configuration Window*.

► To set user gain variables:

1. In the *voicelfMIB*, locate the *voicelfTable* group.
2. Define the following variables:
 - *voicelfUserInputGainOffset*: User input gain offset in dB (from analog to digital).

- *voicelfUserOutputGainOffset*: User output gain offset in dB (from digital to analog).

Values range from -30 dB to +20 dB. However, going above +6 dB may introduce clipping/distortion depending on the country selected. Under -24 dB, you will not have much signal either.

3. Restart the Mediatrix 4100 so that the changes may take effect.

This chapter describes how to perform fax transmissions in clear channel and T.38 with the Mediatrix 4100. You can also set some of these parameters via the web interface, as described in [“Voice & Fax Codecs” on page 79](#).

Introduction

The Mediatrix 4100 handles G3 fax transmissions at speeds up to 14.4 kbps. Automatic fax mode detection is standard on all lines. Real-Time Fax Over UDP with the T.38 protocol stack is also available.

The quality of T.38 fax transmissions depends upon the system configuration, type of call control system used, type of Mediatrix units deployed, as well as the model of fax machines used. Should some of these conditions be unsatisfactory, performance of T.38 fax transmissions may vary and be reduced below expectations.

A fax call works much like a regular voice call, with the following differences:

1. The fax codec may be re-negotiated by using a re-INVITE.
2. The goal of the re-INVITE is to allow both user agents to agree on a fax codec, which is either:
 - a. Clear channel (PCMU/PCMA or G.726) without Echo Cancellation nor Silence Suppression (automatically disabled).
 - b. T.38.
3. Upon fax termination, if the call is not BYE, the previous voice codec is recovered with another re-INVITE.

All lines of the Mediatrix 4100 can simultaneously use the same codec (for instance, T.38), or a mix of any of the supported codecs. Set and enable these codecs for **each** line.

Fax Calling Tone Detection

You can enable the fax calling tone (CNG tone) detection.

► **To enable fax calling tone detection:**

1. In the *dataIffMIB*, set the *dataIffCngToneDetectionEnable* variable to **enable**.

Upon recognition of the CNG tone, the Mediatrix 4100 switches the communication from voice mode to fax mode and the CNG is transferred by using the preferred fax codec.

This option allows for quicker fax detection, but it also increases the risk of false detection.

If you do not want the Mediatrix 4100 to detect the fax calling tone, set the variable to **disable**. In this case, the CNG tone does not trigger a transition from voice to data and the CNG is transferred in the voice channel.

With this option, faxes are detected later, but the risk of false detection is reduced.

CED Fax Tone Detection

You can define the behaviour of the Mediatrix 4100 upon reception of a CED fax tone from the network.

► **To enable CED fax calling tone detection:**

1. In the *dataIbmIB*, set the *dataIbCedFaxToneEnable* variable with the proper value.

Table 161: CED Fax Tone Detection Parameters

Parameter	Description
enable	Upon reception of a CED tone following a sent CNG tone, the unit switches the communication from voice mode to fax mode and the CED is transferred by using the preferred fax codec.
disable	The CED tone does not trigger a transition from voice to data and the CED is transferred in the voice channel.

This configuration has no effect if the *dataIbCngToneDetectionEnable* variable is set to **enable**.

Analog CED Detection Behaviour

You can define the behaviour of the Mediatrix 4100 upon reception of a CED fax tone from the analog port.

► **To define the analog CED detection behaviour:**

1. In the *dataIbmIB*, set the *dataIbAnalogCedDetectionBehavior* variable with the proper value.

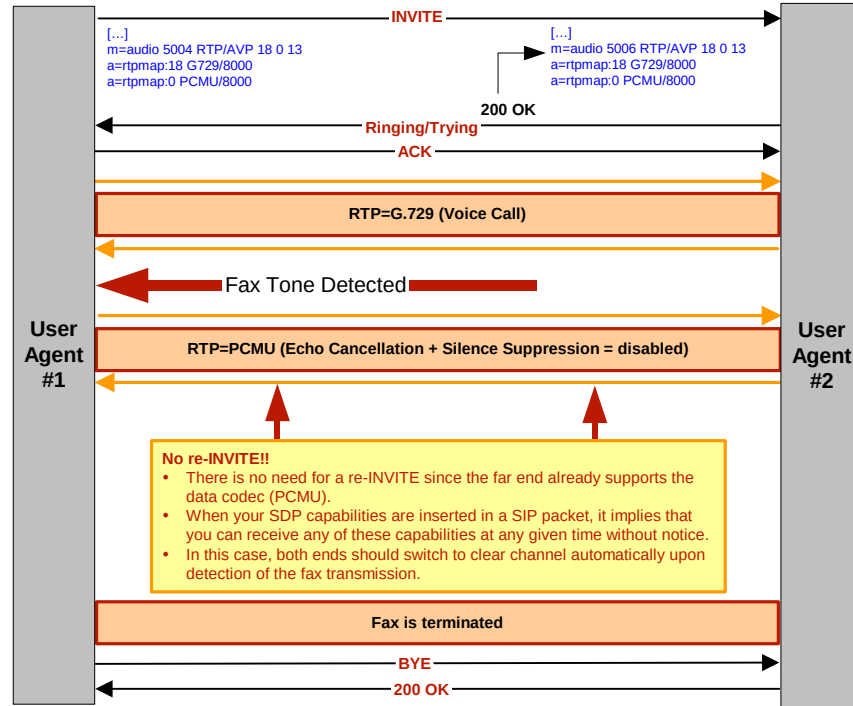
Table 162: Analog CED Detection Behaviour Parameters

Parameter	Description
passthrough	The CED tone does not trigger a transition from voice to data and is transferred in the voice channel. Use this setting when any kind of analog device (i.e.: telephone, fax or modem) can be connected to this port.
faxmode	Upon reception of a CED tone, the unit switches the communication from voice mode to fax mode and the CED is transferred by using the preferred fax codec. Only a fax can then be connected to this port.

Clear Channel Fax

The Mediatrix 4100 can send faxes in clear channel. The following is a clear channel fax call flow:

Figure 83: Clear Channel Fax Call Flow



► **To set a preferred clear channel fax transmission codec:**

1. Set the clear channel codec to use upon detecting a fax tone in the *dataIfClearChannelCodecPreferred* variable.

This variable is used to decide which of the following codecs is preferred, even for voice transmissions:

- PCMU
- PCMA
- G.726 at 32 kbs
- G.726 at 40 kbs



Note: In clear channel, G.726 at 16 kbs and 24 kbs are not available for fax transmission.



Note: If you want to set the G.726 codec at 32 kbs or at 40 kbs as the preferred clear channel codec, you must also select the corresponding G.726 codec as the preferred voice codec as described in ["Preferred Codec" on page 233](#). Otherwise, the Mediatrix 4100 will fail to switch to the G.726 codec for clear channel faxes because G.726 is not negotiated.

- noPreferredCodec

When *noPreferredCodec* is selected and no data-capable codecs are negotiated, data transmission may fail.

It has an impact only if a codec other than PCMU, or PCMA or G.726 is chosen in the *voicelfCodecPreferred* variable (see ["Preferred Codec" on page 233](#)). For instance, if G.729 is the preferred voice codec, then PCMU, and PCMA and G.726 are ordered following the *dataIfClearChannelCodecPreferred* setting.

Clear channel faxes use the negotiated codec, regardless of the setting applied to *dataIfClearChannelCodecPreferred*.

This variable increases the relative priority of the selected codec vs other data-capable codecs. However, the priority of the preferred clear channel codec remains lower than the *voicelfCodecPreferred* variable (see ["Preferred Codec" on page 233](#)).

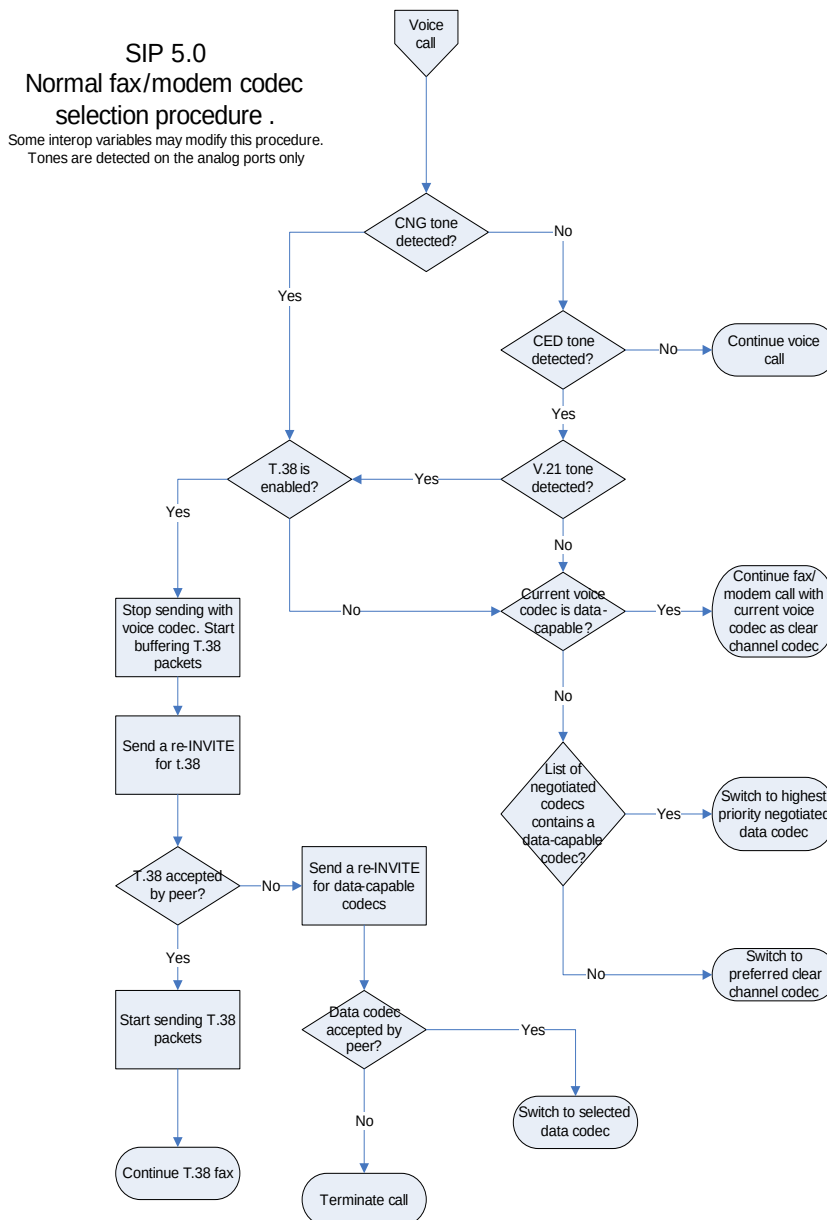
Moreover, when no data-capable codec is part of the list of negotiated codecs, this variable indicates which codec to use when fax or modem tones are detected. However, if the negotiated voice codec is data-capable, the voice codec will be used for data instead of the preferred data codec. See ["Data Codec Selection Procedure" on page 253](#) for more details.

Media5 suggests to use *pcma* if you are located in Europe and *pcmu* anywhere else. However, you should check first which codec is supported in your telephone network.

Data Codec Selection Procedure

The Mediatrix 4100 follows a procedure when selecting data codec. This procedure is the default behaviour of the Mediatrix 4100. Some interop variables may modify this procedure. Tones are detected on the analog ports only.

Figure 84: Data Codec Selection Procedure



T.38 Fax

Standards Supported

- Based on *draft-ietf-sipping-realtimefax-01.txt*
- Recommendation ITU T.38 version 0

T.38 fax relay is a real-time fax transmission; that is, two fax machines communicating with each other as if there were a direct phone line between the two. T.38 is called a fax relay, which means that instead of sending inband fax signals, which implies a loss of signal quality, it sends those fax signals out-of-band in a T.38 payload, so that the remote end can reproduce the signal locally.

The Mediatrix 4100 can send faxes in T.38 mode over UDP or TCP. T.38 is used for fax if both units are T.38 capable; otherwise, transmission in clear channel over G.711 as defined is used (if G.711 μ -law and/or G.711 A-law are enabled). If no clear channel codecs are enabled and the other endpoint is not T.38 capable, the fax transmission fails.



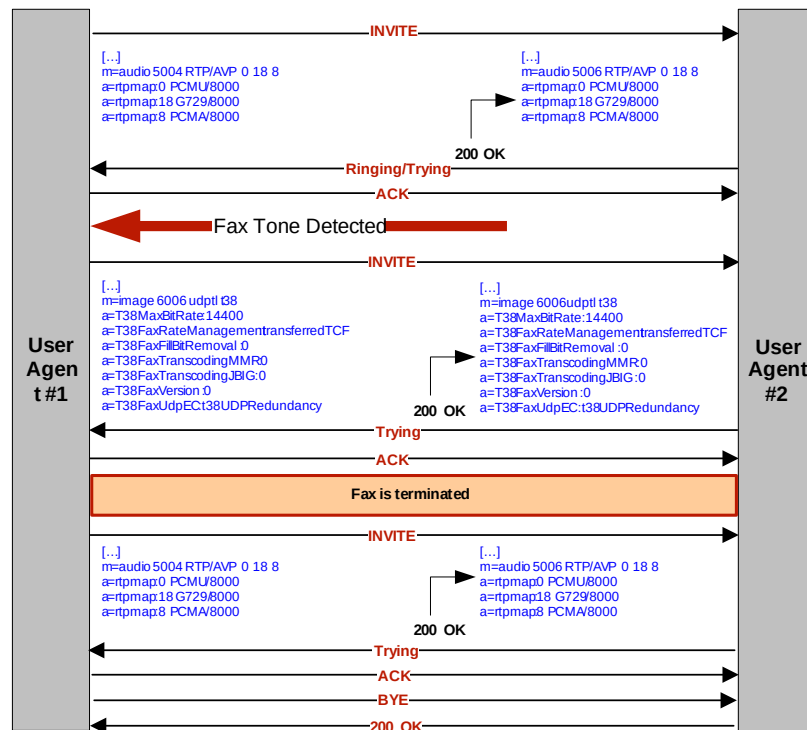
Caution: The Mediatrix 4100 opens the T.38 channel only after receiving the “200 OK” message from the peer. This means that the Mediatrix 4100 cannot receive T.38 packets before receiving the “200 OK”. Based on RFC 3264, the T.38 channel should be opened as soon as the unit sends the “INVITE” message. See [“Fax Issues” on page 355](#) for more details.



In the *Unit Manager Network Administration Manual*, refer to chapter *Ports Parameters*, section *Port Configuration Window*.

The following is a T.38 fax call flow:

Figure 85: T.38 Fax Call Flow



► To set T.38 fax transmission:

- Enable T.38 by setting the `dataIfCodecT38Enable` variable to **enable**.

2. Set the number of redundancy packets sent with the current packet in the *datafT38ProtectionLevel* variable.
This is the standard redundancy offered by T.38. Please see step 3 for additional reliability options for T.38.
Available values range from 1 to 5, 3 being the default value.
3. For additional reliability, define the number of times T.38 packets are retransmitted in the *datafT38FinalFramesRedundancy* variable.
This only applies to the T.38 packets where the PrimaryUDPTL contains the following T.38 data type:
 - HDLC_SIG_END,
 - HDLC_FCS_OK_SIG_END,
 - HDLC_FCS_BAD_SIG_END and
 - T4_NON_ECM_SIG_END
4. Restart the Mediatrix 4100 so that the changes may take effect.

T.38 No-Signal

You can set the Mediatrix 4100 to send no-signal packets during a T.38 fax transmission. The Mediatrix 4100 sends no-signal packets if no meaningful data have been sent for a user-specified period of time.

► To send T.38 no-signal:

1. Set the period, in seconds, at which no-signal packets are sent during a T.38 transmission in the *datafT38NoSignalTimeout* variable.
No-signal packets are sent out if there are no valid data to send.
2. Enable the sending of T.38 no-signal packets by setting the *datafT38NoSignalEnable* variable to **enable**.

Behaviour when Switching from Voice to T.38

You can define the behaviour when the unit starts sending T.38 no-signal packets when being Re-Invited to switch from voice to T.38.

► To define the behaviour:

1. In the *sipInteropMIB*, set the *sipInteropT38NoSignalBehavior* variable to the proper value.

Figure 86: Behaviour when Switching from Voice to T.38 Parameters

Parameter	Description
receivingReInvite	The unit starts sending T.38 no-signal packets when receiving the Re-Invite.
receivingAck	The unit waits until the ACK to its 200 OK has been received before sending T.38 no-signal packets.

This configuration has no effect if the *datafT38NoSignalEnable* variable is set to **disable** (see [“T.38 No-Signal” on page 255](#) for more details).

T.38 INVITE Rejected with 606

You can define the behavior of the Mediatix 4100 when receiving a 606 SIP error response to an INVITE for T.38 fax.

► To define the behavior

1. In the *sipInteropMIB*, set the *sipInteropBehaviorOnT38InviteRejectedWith606* variable to the proper value.

Figure 87: Behaviour when T.38 INVITE is Rejected with 606 Parameters

Parameter	Description
dropCall	The call is dropped by sending a BYE.
usePreviousMedia Negotiation	No re-INVITE is sent and the audio codec from the last successful negotiation is used. For the remainder of the call, T.38 is disabled and fax detection may trigger a switch to a clear channel codec that was available in the last successful negotiation.

Bypass Configuration

The Mediatrix 4100 may have an optional RJ-11 connector used to connect to a standard SCN line, called *Bypass*. It allows its users to maintain telephone services in the event of a power outage or network failure.

Bypass Connector Settings

During normal operation, the SCN line connected to the *Bypass* connector is switched out of the circuit through commuting relays. The *Bypass* connector can be activated by two different conditions:

- ▶ When power is removed from the Mediatrix 4100.
- ▶ When the IP network is down.

If one of these conditions is met, a phone/fax used on FXS connector 1 (Mediatrix 4108/4116) or analog line 1 (Mediatrix 4124) is directly connected to the SCN Bypass line. FXS connector 1 (Mediatrix 4108/4116) or analog line 1 (Mediatrix 4124) stays in Bypass connection until:

- ▶ The error conditions have been cleared.
- ▶ The device connected to it is on-hook and a delay has elapsed.

Standard Bypass

The following describes how to enable/disable the standard Bypass feature.

▶ **To enable the standard Bypass feature:**

1. In the *fxsMIB*, locate the *fxsByPassEnable* variable.
This option enables/disables the bypass service.

Table 163: Bypass Values

Value	Description
disable	The line with the bypass service is never redirected on the bypass line except when there is a power failure.
enable	When the line with the bypass service is unusable (<i>ifAdminUsageState</i> is idle-unusable), it is redirected to the bypass line. When this line becomes usable again (<i>ifAdminUsageState</i> is idle), the redirection is stopped within 10 seconds if the bypass line is unused or 10 seconds after the termination of the call.



Note: The control of the bypass service is only possible when the unit is powered on. When power is off, the bypass service is always enabled.

The default value is **enable**.

This chapter defines how to configure the SIP-specific features to properly use the SIP signalling programs and information defined in Media5' SIP stack.

User Agents

A user agent is a logical entity that can act as both client and server for the duration of a dialog. Each line (also known as endpoint) of the Mediatix 4100 is a user agent.

You can set information for each user agent such as its telephone number and friendly name. This information is used to dynamically create the *To*, *From* and *Contact* headers used in the request the user agent sends. These headers make up the caller ID information that is displayed on telephones/faxes equipped with a proper LCD display. See [“Caller ID Information” on page 177](#) for more details.

Most of the variables related to the user agents are located in tables. You can display and define the information for all lines. You can also use these tables to create/edit five user names and passwords per line. This means that:

- ▶ Rows 1-5 of the table are reserved for line 1.
- ▶ Rows 6-10 of the table are reserved for line 2.
- ▶ etc.

If you want to enter a user name for the second line, you must do so in the sixth row of the table. If you want to enter a user name for the third line, you must do so in the eleventh row of the table, and so on.

Before changing a parameter value, build its corresponding table with your MIB browser's table functionality. Depending on the MIB browser you are using, the tables may not appear the same way.



In the *Unit Manager Network Administration Manual*, refer to chapter *Signalling Protocols Parameters*, section *SIP Configuration Window*.

You can also set these parameters via the web interface, as described in [“SIP User Agent” on page 65](#).

▶ To set basic user agent information:

1. In the *sipMIB*, set the user agent port number in the *sipPort* variable.
The default value is 0. If *sipPort* is set to 0, the default SIP port is used.
2. Define whether or not to override the default proxy home domain used by entering a domain in the *sipDomain* variable.
This value replaces the home domain proxy host as defined in [“Proxy Server” on page 167](#). It is used by the address of record in the *To* and *From* headers.
3. In the *sipUAIfTable* group, set a main user name in the *sipUAMainUsername* variable.
The main user name uniquely identifies this endpoint in the domain, such as a telephone number. It is used to create the *Contact* and *From* headers. The *From* header carries the permanent location (IP address, home domain) where the endpoint is located. The *Contact* header carries the current location (IP address) where the endpoint can be reached. Contact headers are used in two ways:
 - First, contacts are registered to the registrar. This enables callers to be redirected to the endpoint's current location.
 - Second, a contact header is sent along with any request the user agent sends (e.g., INVITE), and is used by the target user agent as a return address for later requests to this endpoint.

You cannot set this field to an empty value. Furthermore, it is reset to 333000X during a factory reset, the X digit being the port number.

4. Set a display name in the *sipUADisplayName* variable.
This is a friendly name for the user agent. It contains a descriptive version of the URI and is intended to be displayed to a user interface.
5. Define a list of other accepted user names in the *sipUAOtherAcceptedUsernames* variable.
This is a list of user names that the endpoint recognizes as its own, but does not register in contacts sent to the registrar. The endpoint only registers the user name in *sipUAMainUsername*.
You can use this variable to add variations on the main user name. For instance, let's say that the main user name is a telephone number, 555-1111. Variations could be to prefix the local area or country code, such as 819-555-1111.
To include more than one user name, separate them with a "," character, such as: user1, user2, 5552222, 18195552222.
6. Restart the Mediatix 4100 so that the changes may take effect.

Home Domain Override

You can override the home domain configuration. The address of record in the REGISTER uses this string instead of the SIP domain as set in the *sipDomain* variable (see ["User Agents" on page 259](#)) or home domain proxy host (variable *sipHomeDomainProxyStaticHost* variable (see ["Proxy Server" on page 167](#) for details).

► To override the home domain configuration:

1. In the *sipInteropMIB*, set the *sipInteropRegisterHomeDomainHostOverride* variable with the proper IP address or domain name.

SIP User Agent Header

Standards Supported	RFC 3261 – SIP: Session Initiation Protocol, section 20.41 (User-Agent)
----------------------------	---

The *User-Agent* header field contains information about the user agent client originating the request. For instance, the information of the *User-Agent* header could be something like the following:

User-Agent: Softphone Beta1.5

You can specify whether or not the Mediatix 4100 sends this information when establishing a communication and customize the header value.

► To enable sending the SIP User Agent header:

1. In the *sipInteropMIB*, set the *sipInteropSendUAHeaderEnable* variable to **enable**.
2. Set the *sipInteropUAHeaderConfig* variable with the proper macro.
The following macros are replaced by their representation:
 - **%version%**: Version of the application.
 - **%mac%**: Unit MAC address (lowercase).
 - **%rev%**: Hardware revision number.
 - **%product%**: Product name.
 - **%%**: A '%' sign.
3. Restart the Mediatix 4100 so that the changes may take effect.

Session Timers

The session timer extension allows to detect the premature end of a call caused by a network problem or a peer's failure by resending an INVITE at every *n* seconds.

A successful response (200 OK) to this INVITE indicates that the peer is still alive and reachable. A timeout to this INVITE may mean that there are problems in the signalling path or that the peer is no longer available. In that case, the call is shut down by using normal SIP means.

- To set Session Timer information:
1.

In the *sipUAI*Table group, set the session timer maximum expiration delay in the *sipUAMaximumSessionExpirationDelay* variable.
This is the suggested maximum time, in seconds, for the periodical session refreshes. It must be equal to or greater than the minimum value. This value is reflected in the *Session-Expires* header.
2.

Set the session timer minimum expiration delay in the *sipUAMinimumSessionExpirationDelay* variable.
This is the minimum value, in seconds, for the periodical session refreshes. It must be equal to or smaller than the maximum value. This value is reflected in the *Min-SE* header.
The *Min-SE* value is a threshold under which proxies and user agents on the signalling path are not allowed to go.
- To disable the Session Timer service:
1.

Set the *sipUAMaximumSessionExpirationDelay* variable to 0.
Increasing the maximum helps to reduce network traffic, but also makes “dead” calls longer to detect.

Session Timer Version

Standards Supported	- draft-ietf-sip-session-timer-08.txt - draft-ietf-sip-session-timer-04.txt (expired)
---------------------	---

You can select the version of the session timer draft that the Mediatrix 4100 uses. Session timer versions other than those provisioned may not work because of backward compatibility issues between the versions.

This feature is currently located under the *mediatrixExperimental* branch of the MIB structure. See [“MIB Structure” on page 135](#) for more details.

The Mediatrix 4100 supports the following session timer versions:

Table 164: Session Timer Versions Supported

Version	Description
sessionTimer04	The Mediatrix 4100 uses the session timer extension as described in the now expired <i>draft-ietf-sip-session-timer-04.txt</i> . Its use is deprecated. You should use this setting for backwards compatibility issues only.
sessionTimer08	The Mediatrix 4100 uses the session timer extension as described in the more recent <i>draft-ietf-sip-session-timer-08.txt</i> . This draft version contains several enhancements over the previous ones, including the use of the <i>Min-SE</i> header. Use this setting if you do not need to interoperate with session timer v4-enabled parties.

- To set the version of session timers supported:
1.

In the *sipInteropMIB*, set the *sipInteropSessionTimersVersion* variable with the proper version.
 - sessionTimer04
 - sessionTimer08

Background Information

The following explains how the session timers are used.

SDP in Session Timer reINVITEs

The reINVITE is sent with the last SDP that was negotiated. Receiving a session timer reINVITE should not modify the connection characteristics.

Relation Between Minimum and Maximum Values

A user agent that receives a *Session-Expires* header whose value is smaller than the minimum it is willing to accept replies a “422 Timer too low” to the INVITE and terminates the call. The phone does not ring.

It is up to the caller to decide what to do when it receives a 422 to its INVITE. Mediatix units will automatically retry the INVITE, with a *Session-Expires* value equal to the minimum value that the user agent server was ready to accept (located in the *Min-SE* header). This means that the maximum value as set in the Mediatix unit might not be followed. This has the advantageous effect of establishing the call even if the two endpoints have conflicting values. Mediatix units will also keep retrying as long as they get 422 answers with different *Min-SE* values.

Who Refreshes the Session?

Re-sending a session timer INVITE is referred to as refreshing the session. Normally, the user agent server that receives the INVITE has the last word on who refreshes. Mediatix units always let the user agent client (caller) perform the refreshes if the caller supports session timers. In the case where the caller does not support session timers, the Mediatix unit assumes the role of the refresher.

Authentication

Standards Supported

Basic and Digest authentication as per RFC 3261

Authentication information allows you to add some level of security to the Mediatix 4100 lines by setting user names and passwords. You can add two types of authentication information:

- ▶ line-specific authentication
- ▶ unit authentication

When a realm requests authentication, the line-specific authentication is tried first, and then the unit authentication if required.

You can also set these parameters via the web interface, as described in [“SIP Authentication” on page 71](#).

Line-Specific Authentication

You can define up to five user names and five passwords for each line of the Mediatix 4100. A line can thus register with five different realms. Keep in mind that:

- ▶ Rows 1-5 of the table where you define the user names and passwords are reserved for line 1.
- ▶ Rows 6-10 of the table where you define the user names and passwords are reserved for line 2.
- ▶ etc.

For instance, to enter a user name for the second line, you must do so in the sixth row of the table. To enter a user name for the third line, you must do so in the eleventh row of the table, and so on.

► **To set line-specific authentication:**

1. In the *sipUAlfAuthenticationTable* group, set the following information:

Table 165: Line-Specific Authentication

Variable	Description
sipUAAuthRealm	When authentication informations are required from users, the realm identifies who requested the information.
sipUAAuthUsername	A string that uniquely identifies this endpoint in the realm, used for authentication purposes. The user name always maps to a password.
sipUAAuthPassword	User password.
sipUAAuthValidateRealm	When set to disable, the current user credentials are valid for any realm. When set to enable, the user credentials are used only for a specific realm set in the sipUAAuthRealm variable.

Unit Authentication

You can define up to five user names and five passwords for the Mediatrix 4100. These user names and passwords apply to all lines of the unit.



In the *Unit Manager Network Administration Manual*, refer to chapter *Signalling Protocols Parameters*, section *SIP Authentication*.

► **To set unit authentication:**

1. In the *sipUnitAuthenticationTable* group, set the following information:

Table 166: Unit-Specific Authentication

Variable	Description
sipUnitAuthRealm	When authentication informations are required from users, the realm identifies who requested the information.
sipUnitAuthUsername	A string that uniquely identifies this endpoint in the realm, used for authentication purposes. The user name always maps to a password.
sipUnitAuthPassword	User password.
sipUnitAuthValidateRealm	When set to disable, the current unit credentials are valid for any realm. When set to enable, the unit credentials are used only for a specific realm set in the <i>sipUnitAuthRealm</i> variable.

Authentication Request Protection

When the Mediatrix 4100 sends an authentication request, you can configure it so that it tries to apply the authentication with integrity protection when this feature is supported by the SIP server.

► **To set the quality of protection:**

1. In the *sipInteropMIB*, specify the quality of protection the SIP User Agent should apply to its authentication request in the *sipInteropAuthenticationQop* variable.

The following values are supported:

Table 167: Quality of Protection

Parameter	Description
auth	The SIP User Agent applies authentication only. This is the default value.

Table 167: Quality of Protection (Continued)

Parameter	Description
auth-int	The SIP User Agent applies authentication with integrity protection (see RFC 2617).

- Restart the Mediatix 4100 so that the changes may take effect.

SIP Trusted Sources

You can configure the Mediatix 4100 so that it only accepts SIP messages coming from one of six trusted IP addresses. All other SIP messages are ignored. This source address validation takes place at the IP level and does not depend on the SIP header or body.

► **To configure the SIP trusted sources feature:**

- In the *sipMIB*, set the *sipTrustedSourcesIpAddress* variable with the IP address of a trusted source of SIP messages.
You can enter up to six addresses.
- Enable the SIP trusted sources feature by setting the *sipTrustedSourcesEnable* to **enable**.
If you set the variable to **disable**, the Mediatix 4100 does not validate the source address of SIP messages at the IP level.

NAT Traversal

The Mediatix 4100 may be used in a private domain that is not directly connected to the IP network. For instance, this may be the case for ITSP (Internet Telephony Service Provider) clients that have a small private network. This private network is connected to the public IP network through the NAT (Name Address Translation) technology.

Currently only one Mediatix unit can be deployed behind a standard NAT.

You can configure the Mediatix 4100 with the public IP address of the NAT system, which allows to reach the unit. SIP packets sent by the Mediatix 4100 contain the NAT address configured as SIP contact. If the NAT service is not activated, the real IP address of the Mediatix 4100 is used.

This method is recommended when the public IP address of the NAT system is static or does not change regularly since it would cause downtime until it is changed manually.

Network Address Translation

Network Address Translation (NAT) is a method of connecting multiple computers to the Internet (or any other IP network) by using one IP address. This allows home users and small businesses to cheaply and efficiently connect their network to the Internet. The basic purpose of NAT is to multiplex traffic from the internal network and present it to the Internet as if it was coming from a single computer having only one IP address.

NAT automatically provides firewall-style protection without any special set-up because it only allows connections originating on the inside network. This means, for instance, that an internal client can connect to an outside FTP server, but an outside client cannot connect to an internal FTP server because it would have to originate the connection, and NAT does not allow that.

Mediatrix 4100 Configuration

This section describes how to activate the NAT service of the Mediatrix 4100.

► To activate the NAT service:

1. In the *ipAddressConfig* folder, set the *localHostWanAddressSelectConfigSource* variable to **static**.
2. Enter the public IP address of the NAT system in the *localHostStaticWanAddress* variable.
This is the public IP address used as Contact address by outgoing SIP packets crossing a NAT system.

NAT System Configuration

You must configure the NAT system to some degree. The configuration required depends on the type of NAT system you are using, but this usually involves port forwarding configuration.

SIP Transport Type

Standards Supported

RFC 3261 – SIP: Session Initiation Protocol

You can globally set the transport type for all the lines of the Mediatrix 4100 to either UDP (User Datagram Protocol) or TCP (Transmission Control Protocol). The Mediatrix 4100 will include its supported transports in its registrations.

Please note that RFC 3261 states the implementations must be able to handle messages up to the maximum datagram packet size. For UDP, this size is 65,535 bytes, including IP and UDP headers. However, the maximum datagram packet size the Mediatrix 4100 supports for a SIP request or response is 5120 bytes excluding the IP and UDP headers. This should be enough, as a packet is rarely bigger than 2500 bytes.

You can also set these parameters via the web interface, as described in [“SIP Transport Type” on page 68](#).

► To set the transport type:

1. In the *sipMIB*, set the priority order of the transport in the *sipTransportQValue* variable.
A *qvalue* parameter is added to each contact. This only applies if the transport-specific registration is enabled.
The *qvalue* gives each transport a weight, indicating the degree of preference for that transport. A higher value means higher preference.
The format of the *qvalue* string must follow the RFC 3261 ABNF (a floating point value between 0.000 and 1.000). If you specify an empty string, no *qvalue* is set in the contacts.
Because this variable is located in a table, you can have a different value for each line.
2. Enable the transport by setting the *sipTransportEnable* variable to **enable**.
The UDP and TCP transport types are located in a table with two rows – one for each type. You can enable/disable a type for the unit.
If the TCP transport type is not used, Media5 strongly suggests to disable it.
3. Enable the transport registration by setting the *sipTransportRegistrationEnable* variable to **enable**.
The Mediatrix 4100 includes its supported transports in its registrations. It registers with one contact for each transport that is currently enabled. Each of these contacts contains a “transport” parameter.

This is especially useful for a system where there are no SRV records configured to use a predefined transport order for receiving requests. When sending a request, the unit either follows the SRV configuration, or, if not available, any transport parameter received from a redirection or from a configured SIP URL. See [“Chapter 10 - DNS SRV Configuration” on page 173](#) for more details.



Note: If the Mediatrix 4100 has the following configuration:

- the *sipTransportRegistrationEnable* variable is set to **disable**
- the UDP transport type is disabled
- the TCP transport type is enabled

The Mediatrix 4100 will not work properly unless the SIP server uses the TCP transport type by default.

This is also true if the Mediatrix 4100 has the TCP transport disabled and the UDP transport enabled. In this case, the Mediatrix 4100 will not work properly unless the SIP server uses the UDP transport protocol by default.

4. Restart the Mediatrix 4100 so that the changes may take effect.

Transport Parameter

You can define whether the Mediatrix 4100 must include its supported transport in all SIP messages that have the *Contact* header, except for the REGISTER message. See [“SIP Transport Type” on page 265](#) for details on how to include transport parameters in the REGISTER message.

If enabled, then the Mediatrix 4100 will send SIP messages with the “transport” parameter in the *Contact* header set to the currently supported transport type.

► To include the supported transport in the contact header:

1. In the *sipMIB*, indicate whether or not the unit must include its supported transport in the *Contact* header in the *sipTransportContactEnable* variable.

Available values are *enable* and *disable*. If you set the variable to enable, the transport parameter is either set to:

- *transport=tcp* when TCP is enabled and UDP is disabled
- *transport=udp* when UDP is enabled and TCP disabled
- no transport parameter when both TCP and UDP are enabled

UDP Source Port Behaviour

You can configure if the Mediatrix 4100 always uses the same local port (the port on which it is listening for incoming packets) when sending SIP traffic over UDP. This is called symmetric UDP source port. Symmetric UDP ports are sometimes needed to traverse NAT/Firewall devices.

When changing this setting, all destinations are automatically sent out of the penalty box, when applicable.

This feature is currently located under the *mediatrixExperimental* branch of the MIB structure. See [“MIB Structure” on page 135](#) for more details.

► To set the UDP source port behaviour:

1. In the *sipInteropMIB*, set the *sipInteropSymmetricUdpSourcePortEnable* variable to **enable**.

The SIP signalling sent over UDP originates from the same port as the port on which the user agent is listening (see [“User Agents” on page 259](#) for details). ICMP messages are not processed, which means that unreachable targets will take longer to detect.

If you set the variable to **disable**, the SIP signalling over UDP uses a randomly-generated originating port. ICMP errors are processed correctly.

2. Restart the Mediatrix 4100 so that the changes may take effect.

SIP Penalty Box

The penalty box feature is used to “quarantine” a given host which address times out. During that time, the address is considered as “non-responding” for all requests.

This feature is most useful when using multiple servers and some of them are down. It ensures that users wait a minimal period of time before trying a secondary host.

You can also set these parameters via the web interface, as described in [“SIP Penalty Box” on page 67](#).

Penalty Box vs Transport Types

Media5 recommends to use this feature with care when supporting multiple transports (see [“SIP Transport Type” on page 265](#) for more details) or you may experience unwanted behaviours.

When the Mediatrix 4100 must send a packet, it retrieves the destination from the packet. If the destination address does not specify a transport to use and does not have a DNS SRV entry that configures which transport to use, then the Mediatrix 4100 tries all transports it supports, starting with UDP. If this fails, it tries with TCP. The unit begins with UDP because all SIP implementations must support this transport, while the mandatory support of TCP was only introduced in RFC 3261.



Note: It is not the destination itself that is placed in the penalty box, but the combination of address, port and transport. When a host is in the penalty box, it is never used to try to connect to a remote host unless it is the last choice for the Mediatrix 4100 and there are no more options to try after this host.

Let’s say for instance that the Mediatrix 4100 supports both the UDP and TCP transports. It tries to reach endpoint “B” for which the destination address does not specify a transport and there is no DNS SRV entry to specify which transports to use in which order. It turns out that this endpoint “B” is also down. In this case, the Mediatrix 4100 first tries to contact endpoint “B” via UDP. After a timeout period, UDP is placed in the penalty box and the unit then tries to contact endpoint “B” via TCP. This fails as well and TCP is also placed in the penalty box.

Now, let’s assume endpoint “B” comes back to life and the Mediatrix 4100 tries again to contact it before UDP and TCP are released from the penalty box. First, the unit tries UDP, but it is currently in the penalty box and there is another transport left to try. The Mediatrix 4100 skips over UDP and tries the next target, which is TCP. Again, TCP is still in the penalty box, but this time, it is the last target the Mediatrix 4100 can try, so penalty box or not, TCP is used all the same to try to contact endpoint “B”.

There is a problem if endpoint “B” only supports UDP (RFC 2543-based implementation). Endpoint “B” is up, but the Mediatrix 4100 still cannot contact it: with UDP and TCP in the penalty box, the unit only tries to contact endpoint “B” via its last choice, which is TCP.

The same scenario would not have any problem if the penalty box feature was disabled. Another option is to disable TCP in the Mediatrix 4100, which makes UDP the only possible choice for the unit and forces to use UDP even if it is in the penalty box.

You must fully understand the above problem before configuring this feature. Mixing endpoints that do not support the same set of transports with this feature enabled can lead to the above problems, so it is suggested to either properly configure SRV records for the hosts that can be reached or be sure that all hosts on the network support the same transport set before enabling this feature.

Penalty Box Configuration

The following steps describe how to configure the penalty box feature.

► To set the penalty box feature:

1. In the *sipMIB*, locate the *sipPenaltyBox* group.
2. Set the amount of time, in seconds, that a host spends in the penalty box in the *sipPenaltyBoxTime* variable.

Changing the value does not affect IP addresses that are already in the penalty box. The *sipPenaltyBoxTime* only affects new entries in the penalty box.

3. Enable the SIP penalty box feature by setting the *sipPenaltyBoxEnable* variable to **enable**.
The penalty box is always “active”. This means that even if the feature is disabled, IP addresses are marked as invalid, but they are still tried. This has the advantage that when the feature is enabled, IP addresses that were already marked as invalid are instantly put into the penalty box.

Registration Parameters

The following describes registration parameters and behaviours you can configure.

Refreshing Registration

You can refresh the registration, i.e., commit the changes you have done to the registration. When refreshing the registration, all enabled endpoints unregister themselves from the previous registrar and send a new registration to the current registrar with the current parameters.

Variables whose modification require a registration refresh are:

- ▶ *sipRegistrarStaticHost*
- ▶ *sipRegistrarStaticPort*
- ▶ *sipUAMainUsername*
- ▶ *sipUADisplayName*
- ▶ *sipServerSelectConfigSource*
- ▶ *sipTransportRegistrationEnable*
- ▶ *sipTransportEnable* (if *sipTransportRegistrationEnable* is enabled)
- ▶ *sipTransportQValue* (if *sipTransportRegistrationEnable* is enabled)

▶ To refresh the registrations:

1. In the *sipMIB*, set the *sipRegistrationCmdRefresh* variable with the proper behaviour.
The following values are available:
 - noOp: No operation.
 - refresh: Refresh registrations.
2. Define the time, in seconds, at which a registered unit begins updating its registration before the registration expiration in the *sipReRegistrationTime* variable.
For instance, if the variable is set to 43 and the registration lasts one hour, the unit will send new REGISTER requests 59 minutes and 17 seconds after receiving the registration acknowledgement (43 seconds before the unit becomes unregistered).



Note: Normally, the Mediatrix 4100 cannot make or receive calls until the REGISTER has completed successfully. Because the timeout for a SIP transaction in UDP is 32 seconds, it is possible to have an ongoing re-REGISTER transaction at the same moment that the registration itself expires. This could happen if *sipReRegistrationTime* is set to a value lower than 32.

In that case, the user agent becomes unregistered, and will become registered again only when the re-REGISTER request is answered with a positive response from the server. See [“Unregistered Line Behaviour” on page 224](#) for a workaround if the unit cannot make calls during that period.

This value **MUST** be lower than the value of the Expires header of the contact in the 200 OK response to the REGISTER, otherwise the unit rapidly sends REGISTER requests continuously.

Registration Expiration

The SIP protocol allows an entity to specify the “expires” parameter of a contact in a REGISTER request. The server can return this “expires” parameter in the 200 OK response or select another “expires”. In the REGISTER request, the “expires” is a suggestion the entity makes.

The “expires” parameter indicates how long, in seconds, the user agent would like the binding to be valid.

You can configure the “expires” parameter the Mediatrix 4100 sends.

► **To configure the registration expiration:**

1. In the *sipMIB*, set the *sipRegistrationProposedExpirationValue* variable with the suggested expiration delay, in seconds, of a contact in the REGISTER request.
Available values are from 1 s to 86,400 s (one day).
This value does not modify the time before a re-REGISTER.
 - The time is the “expires” of the contact in the 200 OK response to the REGISTER request minus the value set in the *sipReRegistrationTime* variable.
 - If the “expires” of the contact in the 200 OK response to the REGISTER is not present or not properly formatted, then the time is the default registration expires minus the value set in the *sipReRegistrationTime* variable.

See [“Refreshing Registration” on page 268](#) for more details.

Setting the variable to **0** disables the expiration suggestion.

Default Registration Expiration

Standards Supported	RFC 3261 – SIP: Session Initiation Protocol, section 20.41 (User-Agent)
----------------------------	---

RFC 3261 specifies that, when the “expires” parameter or the “Expires” header is missing or not properly formatted for a contact of the 200 OK in response to a REGISTER request, the unit must use a default registration expiration value of 3600 s.

You can configure the value of the default registration expiration.

► **To configure the default registration expiration:**

1. In the *sipInteropMIB*, set the *sipInteropDefaultRegistrationExpiration* variable with the default registration expiration.
The delay before a re-REGISTER is the value set in the *sipInteropDefaultRegistrationExpiration* variable minus the value set in the *sipReRegistrationTime* variable. See [“Refreshing Registration” on page 268](#) for more details.
The recommended value in RFC 3261 (section 10.2) is 3600 seconds.
2. Restart the Mediatrix 4100 so that the change may take effect.

Publication Parameters

Standards Supported	• RFC 3863 – Presence Information Data Format (PIDF) • RFC 3903 – Session Initiation Protocol (SIP) Extension for Event State Publication
----------------------------	--

The following describes publication parameters and behaviours you can configure.

Refreshing Publications

You can refresh the publications, i.e., commit the changes you have done to the publications. When refreshing the publications, all enabled endpoints unpublish themselves from the previous Presence Compositor and send a new publication to the current Presence Compositor with the current parameters.

Variables whose modification require a registration refresh are:

- *sipPresenceCompositorStaticHost*
- *sipPresenceCompositorStaticPort*

► **To refresh the publications:**

1. In the *sipMIB*, set the *sipPublicationCmdRefresh* variable with the proper behaviour.
The following values are available:
 - noOp: No operation.
 - refresh: Refresh publications.
2. Define the time, in seconds, at which the User Agent begins refreshing its publication before the publication expiration in the *sipPublicationRefreshTime* variable.
A publication is valid for a period of time specified by the Presence Compositor. The User Agent is then responsible for refreshing its previously established publications before their expiration interval has elapsed.
For instance, if the publication lasts 60 minutes and this variable is set to 43, the unit sends new PUBLISH requests 59 minutes and 17 seconds after the reception of the publication acknowledgement (43 seconds before the end of the publication period).
Setting this variable to 0 means that the User Agent falls into the 'unpublished' state BEFORE sending the refreshing PUBLISH request.

Publications Expiration

The SIP protocol allows an entity to specify the “expires” parameter of a contact in a PUBLISH request. The server can return this “expires” parameter in the 200 OK response or select another “expires”. In the PUBLISH request, the “expires” is a suggestion the entity makes.

The “expires” parameter indicates how long, in seconds, the user agent would like the binding to be valid. You can configure the “expires” parameter the Mediatrix 4100 sends.

► **To configure the publications expiration:**

1. In the *sipMIB*, set the *sipPublicationProposedExpirationValue* variable with the suggested expiration delay, in seconds, of a publication in the PUBLISH request.
Keep in mind that this is only a suggestion and that servers will decide the publication time following local policy.
Available values are from 1 s to 86,400 s (one day).
This value does not modify the delay before a re-PUBLISH.
 - The delay is the value of the Expires header in the 200 OK response to the PUBLISH minus the value set in the *sipPublicationRefreshTime* variable.
 - If the Expires header in the 200 OK response to the PUBLISH is not present or badly formatted, then the delay is the value of the variable *sipInteropDefaultPublicationExpiration* minus the value set in the *sipPublicationRefreshTime* variable.

See [“Refreshing Publications” on page 269](#) for more details.
Setting the variable to 0 disables the expiration suggestion.

Default Publication Expiration

You can configure the value of the default registration expiration.

► **To configure the default publication expiration:**

1. In the *sipInteropMIB*, set the *sipInteropDefaultPublicationExpiration* variable with the default publication expiration.
The delay before a re-PUBLISH is the value set in the *sipInteropDefaultPublicationExpiration* variable minus the value set in the *sipPublicationRefreshTime* variable. See [“Refreshing Publications” on page 269](#) for more details.
2. Restart the Mediatrix 4100 so that the change may take effect.

Interop Parameters

The interop parameters allow the Mediatrix 4100 to properly work, communicate, or connect with specific IP devices.

Call Transfer Capacity

The following parameters allow you to define how the Mediatrix 4100 handles call transfers.

Call Transfer Version

Standards Supported	• draft-ietf-sip-cc-transfer-05.txt • draft-ietf-sip-cc-transfer-02.txt (expired) • draft-ietf-sip-refer-02.txt • draft-ietf-sipping-cc-transfer-01.txt • RFC 3515 - The Session Initiation Protocol (SIP) (Refer Method)
----------------------------	---

You can select the version of the transfer draft that the Mediatrix 4100 uses. The provisioned version is used for initiating transfers and receiving them. Transfer versions other than those provisioned do not work.

Table 168: Call Transfer Versions Supported

Version	Description
transfer02	The Mediatrix 4100 executes transfers by using the methods described in the now expired <i>draft-ietf-sip-cc-transfer-02.txt</i> . Its use is deprecated and you should use this setting for backward compatibility issues only.
transfer05UsingRefer02	The Mediatrix 4100 executes transfers by using the methods described in the more recent <i>draft-ietf-sip-cc-transfer-05.txt</i> . This draft version contains several enhancements over the previous ones. Among others, it is possible to use the <i>Replaces</i> header to provide a more seamless attended transfer to the user. This method also uses <i>draft-ietf-sip-refer-02.txt</i> . Use this setting if you do not need to interop with transfer02-enabled parties. See “Replaces Configuration Setting” on page 272 for more details.
sippingTransfer01UsingReferRfc3515	The Mediatrix 4100 executes transfers by using the methods described in <i>draft-ietf-sipping-cc-transfer-01.txt</i> . This draft version is more recent than Transfer02 and Transfer05UsingRefer02. This method also uses the <i>RFC 3515 - The Session Initiation Protocol (SIP) Refer Method</i> .

This feature is currently located under the *mediatrixExperimental* branch of the MIB structure. See [“MIB Structure” on page 135](#) for more details.

► **To set the version of transfer supported:**

1. In the *sipInteropMIB*, set the *sipInteropTransferVersion* variable with the proper version.
 - transfer02
 - transfer05UsingRefer02
 - sippingTransfer01UsingReferRfc3515

Replaces Configuration Setting

You can configure how to use the *Replaces* header mechanism used in a transfer. When supported by the target of the transfer, the *Replaces* header mechanism ensures a more seamless transfer by permitting the initiating party to effectively replace a current call by another instead of disconnecting the call to be replaced and creating a second call. This allows you to control how the Mediatix 4100 interoperates with other vendor's products and older Mediatix units.

This feature is currently located under the *mediatrixExperimental* branch of the MIB structure. See "[MIB Structure](#)" on page 135 for more details.

- To set Replaces configuration:
1. In the *sipInteropMIB*, set the Replaces configuration in the *sipInteropReplacesConfig* variable. You have the following choices:

Table 169: Replaces Configuration

Configuration	Description
doNotUseReplaces	The <i>Replaces</i> header is not used.
useReplacesWithRequire	The <i>Replaces</i> header is used. It can be seen in the <i>Refer-To</i> header of the REFER request sent by the transferor. It can also be seen in the INVITE sent by the transferee. The target that supports <i>Replaces</i> uses its information to merge the new INVITE with an existing call specified in the <i>Replaces</i> header. The transferee requires to use the replaces extension for proper completion of the transfer. If the target of the transfer does not support the replaces extension, the Mediatix 4100 retries the transfer using replaces by reversing the roles of the target and the transferee (by resending the REFER to the initial target instead of the initial transferee). As a last resort (if none of the participants supports replaces), the transfer is carried out without using the replaces extension.
useReplacesNoRequire	The <i>Replaces</i> header is used. It can be seen in the <i>Refer-To</i> header of the REFER request sent by the transferor. It can also be seen in the INVITE sent by the transferee. The target that supports <i>Replaces</i> uses its information to merge the new INVITE with an existing call specified in the <i>Replaces</i> header. This disables the transfer fallback. The replaces information is still present, but no check is made that it is effectively used to complete the transfer.

Replaces Version

Standards Supported	• sip-replaces-01 draft • sip-replaces-03 draft
---------------------	---

You can select the version of the *ietf-sip-replaces* draft to which the Mediatix 4100 must conform. The provisioned version affects the way blind transfers are executed.

This feature is currently located under the *mediatrixExperimental* branch of the MIB structure. See [“MIB Structure” on page 135](#) for more details.

Table 170: Replaces Versions Supported

Version	Description
replaces01	The Transferor can use a REFER with Replaces when proceeding to initiate a blind transfer. This results in the Transferee including a Replaces header in its INVITE to the Transfer Target.
replaces03	When initiating a blind transfer, the Transferor first CANCELS its call with the Target and then issues a REFER without Replaces to the Transferee. Note: A side effect is that the phone will stop ringing and start again.

► **To set the version of Replaces supported:**

- In the *sipInteropMIB*, set the *sipInteropReplacesVersion* variable with the proper version.
 - replaces01
 - replaces03

Transmission Timeout

Standards Supported	- RFC 2543bis – SIP: Session Initiation Protocol - RFC 3261 – SIP: Session Initiation Protocol
----------------------------	---

If a DNS SRV answer contains more than one entry, the Mediatrix 4100 will try these entries if the entry initially selected does not work. You can configure the maximum time, in seconds, to spend waiting for answers to messages, from a single source. Retransmissions still follow the algorithm proposed in *RFC 2543bis*, but the total wait time can be overridden by using this feature.

For example, if you are using DNS SRV and more than one entry is present, this timeout is the time it takes before trying the second entry.

This feature is currently located under the *mediatrixExperimental* branch of the MIB structure. See [“MIB Structure” on page 135](#) for more details.

► **To set the transmission timeout:**

- In the *sipInteropMIB*, locate the *sipInteropTransmissionTimeout* variable.
- Set the timeout value.
Available values are from 1 to 32 seconds.

Max-Forwards Header

Standards Supported	RFC 3261 – SIP: Session Initiation Protocol
----------------------------	---

You can configure whether the Mediatrix 4100 inserts the *Max-Forwards* header into sent requests, as per RFC 3261. Max-Forwards serves to limit the number of hops a request can make on the way to its destination. It consists of an integer that is decremented by one at each hop. If the Max-Forwards value reaches 0 before the request reaches its destination, it will be rejected with a “483 (Too Many Hops)” error response.

This feature is currently located under the *mediatrixExperimental* branch of the MIB structure. See [“MIB Structure” on page 135](#) for more details.

► **To insert the Max-Forwards header into SIP requests:**

- In the *sipInteropMIB*, set the *sipInteropMaxForwardsValue* variable to the value you want.
Any positive value means that the *Max-Forwards* header is inserted into sent requests. The default value is **70**.
- Restart the Mediatrix 4100 so that the change may take effect.

► **To disable inclusion of this header in SIP requests:**

1. In the *sipInteropMIB*, set the *sipInteropMaxForwardsValue* variable to **-1**.
2. Restart the Mediatix 4100 so that the change may take effect.

Referred-By Field

The SIP REFER method allows the referrer to provide information about the reference to the refer target by using the referee as an intermediary. The mechanism for carrying the referrer's identity, expressed as a SIP URI, is the *Referred-By* header.

You can configure the *Referred-By* field used in a SIP REFER request to decide whether it contains the permanent URL provided by the SIP stack or the address of record used when the unit registered.

► **To configure the *Referred-By* field:**

1. In the *sipInteropMIB*, set the *sipInteropReferredByConfig* variable to the value you want.

Table 171: Referred-By Field Parameters

Parameter	Description
useSipStackDefault	The SIP stack populates the <i>Referred-By</i> header field.
useLocalUrl	Uses the local URL to populate the <i>Referred-By</i> header field.

Direction Attributes in a Media Stream

The Mediatix 4100 allows you to define various direction attributes pertaining to the media stream.

Direction Attribute

Standards Supported	• RFC 2543 – SIP: Session Initiation Protocol • RFC 3264 – An Offer/Answer Model with Session Description Protocol (SDP)
----------------------------	---

You can define:

- if the SDP direction attribute is present in the initial INVITE sent by the Mediatix 4100
- whether or not the direction attribute present in the SDP received from the peer is ignored

► **To define if the direction attribute is present:**

1. In the *sipInteropMIB*, set the *sipInteropSdpDirectionAttributeEnable* variable to the proper value.

Table 172: SDP Direction Attribute

Parameter	Description
disable	No direction attribute is present in the SDP sent by the Mediatix 4100. The Mediatix 4100 ignores any direction attribute found in the SDP received from the peer. To put an endpoint on hold, a SDP containing a connection address of "0.0.0.0" is sent. The method to put a session on hold is in conformance with RFC 2543.

Table 172: SDP Direction Attribute (Continued)

Parameter	Description
enable	The Mediatix 4100 always sends the direction attribute in the SDP of the initial INVITE. The initial handshake determines whether or not the peer supports the direction attribute. - If the direction attribute is present in the SDP received from the peer, the Mediatix 4100 sends the direction attribute in the SDP for the remainder of the session. - If the direction attribute is not present in the SDP received from the peer, the Mediatix 4100 does not send the direction attribute in the SDP for the remainder of the session. If present in the SDP, the direction attribute is preferred over the connection address to transmit session modification information. This method is in conformance with RFC 3264.

When Putting a Call on Hold

Standards Supported	RFC 3264 – An Offer/Answer Model with Session Description Protocol (SDP)
----------------------------	--

The Mediatix 4100 can provide the direction attribute and the meaning of the connection address “0.0.0.0” sent in the SDP when putting an endpoint on hold.

This configuration has no effect if the *sipInteropSdpDirectionAttributeEnable* variable is set to **Disable** (see [“Direction Attribute” on page 274](#) for more details).

See [“Call Hold” on page 306](#) for more details on holding calls.

► To define the direction attribute when putting a call on hold:

1. In the *sipInteropMIB*, set the *sipInteropOnHoldSdpStreamDirection* variable to the proper value.

Table 173: Direction Attributes

Parameter	Description
inactive	The stream is put on hold by marking it as <i>inactive</i> . This is the default value. This setting should be used for backward compatibility issues.
sendonly	The stream is put on hold by marking it as <i>sendonly</i>. This method allows the Mediatix 4100 to be in conformance with RFC 3264.

Answering a Hold Offer with the Direction Attribute “sendonly”

Standards Supported	RFC 3264 – An Offer/Answer Model with Session Description Protocol (SDP)
----------------------------	--

You can define how to set the direction attribute and the connection address in the SDP when answering a hold offer with the direction attribute “sendonly”.

► To define the behaviour with the “sendonly” direction attribute:

1. In the *sipInteropMIB*, set the *sipInteropOnHoldAnswerSdpStreamDirection* variable to the proper value.

Table 174: “sendonly” Direction Attribute

Parameter	Description
inactive	The stream is marked as inactive and the connection address is set to '0.0.0.0'.
recvonly	If the stream is currently active or receive only, it is marked as recvonly and the connection address is set to the IP address of the unit. If the stream is currently send only or inactive, it is marked as inactive and the connection address is set to '0.0.0.0'. This method is in conformance with RFC 3264.

In both cases, no direction attribute is present in the SDP if the *sipInteropSdpDirectionAttributeEnable* variable is set to **Disable** (see [“Direction Attribute” on page 274](#) for more details).

Allowing Multiple Active Media in Answer

You can define the behaviour of the Mediatix 4100 when answering a request offering more than one active media.

► To allow multiple active media in answer:

1. In the *sipInteropMIB*, set the *sipInteropAllowMultipleActiveMediaInAnswer* variable to the proper value.

Figure 88: Allow Multiple Active Media in Answer

Parameter	Description
disable	The answer contains only one active media. The media specified as active in the answer is the top-most matching one in the offer. Other media are set to inactive.
enable	Each matching active media in the offer is specified as active in the answer. Other media are set to inactive

Local Ring Behaviour on Provisional Response

You can set the Mediatix 4100 so that it starts or not the local ring upon receiving a “18x Provisional” response without SDP.

This setting does not affect the behaviour when the “18x Provisional” response contains SDP, which allows to establish an early media session before the call is answered.

► To define the local ring behaviour on provisional response:

1. In the *sipInteropMIB*, set the *sipInteropLocalRingOnProvisionalResponse* variable to the proper value.

Figure 89: Local Ring Behaviour

Parameter	Description
disable	The local ring is not started on a “18x Provisional” response without SDP, except for a “180 Ringing” message. This is the default value. The 180 Ringing message is a provisional or informational response used to indicate that the INVITE message has been received by the user agent and that alerting is taking place. Note: Using this default value means you are implementing a behaviour that is different from previous versions of the Mediatrix 4100 application.
enable	The local ring is started on any “18x Provisional” response without SDP.

SIP Credential

You can configure how the Mediatrix 4100 reuses the credential in different transactions of the same call or registration. For instance, it may be required that a new SIP request does not reuse the credential negotiated in the previous transaction of the same call or registration. For example, a re-INVITE will not reuse the credential of the INVITE but will be challenged.

► To set the Mediatrix 4100 not to reuse the SIP credential:

1. In the *sipInteropMIB*, set the *sipInteropReuseCredentialEnable* variable to **disable**.
If you set this variable to **enable** (which is the default value), the Mediatrix 4100 reuses the credential negotiated in previous transactions.

Branch Parameter Settings

The following are settings related to the Branch parameter.

Branch Matching Method

Standards Supported	• RFC 2543 – SIP: Session Initiation Protocol, section 10.1.2 • RFC 3261 – SIP: Session Initiation Protocol, section 8.1.1.7
----------------------------	---

You can configure the method used to match incoming SIP packets with a branch. A branch could be described as a link that allows to match a response to a request.

► **To configure the branch matching method:**

1. In the *sipInteropMIB*, set the *sipInteropBranchMatchingMethod* variable with the proper method to use.

Table 175: Branch Matching Method

Method	Description
rfc2543	Follows the method described in RFC 2543 (section 10.1.2). Responses are mapped to requests by the matching <i>To</i> , <i>From</i> , <i>Call-ID</i> , and <i>CSeq</i> headers and the branch parameter of the first <i>Via</i> header.
rfc3261	Follows the method described in RFC 3261 (section 8.1.1.7). A <i>Via</i> is inserted into the request and the <i>Via</i> header field value must contain a branch parameter. This parameter is used to identify the transaction created by that request. It is used by both the client and the server. The branch ID is used to facilitate its use as a transaction ID. It must always begin with the characters "z9hG4bK". If this is not the case, the Mediatix 4100 uses the branching method as described in RFC 3261, section 17.2.3.

Transaction Matching Procedure

You can configure the use of the *Via* branch behaviour for incoming CANCEL requests. You can specify whether the SIP stack's transaction matching procedure ignores the branch parameter of the *Via* header field in CANCEL requests with no *To* tag.

► **To configure the use of the *Via* branch behaviour for CANCEL requests:**

1. In the *sipInteropMIB*, set the *sipInteropIgnoreViaBranchIdInCancelEnable* variable with the proper behaviour.

Table 176: Via Branch Behaviour

Method	Description
disable	The transaction matching procedure behaves according to section 17.2.3 of RFC 3261. This is the default value.
enable	The branch parameter is not used as a transaction matching criterion for CANCEL requests with no <i>To</i> tag.

Ringing Response Code

You can configure the response code sent back when the line starts ringing.

► **To configure the response code sent back:**

1. In the *sipInteropMIB*, set the *sipInteropRingingResponseCode* variable with the proper code to send back.

Table 177: Ringing Response Code

Method	Description
send180Ringing	The Mediatrix 4100 sends out a "180 Ringing" response without a body. In this case, the ringback the caller hears is generated by his own unit upon receiving the message. This is the default value.
send183WithSdp	The Mediatrix 4100 returns a "183 Session Progress" packet with SDP (needed if the endpoint is required to generate ringback on connection). In this case, the RTP channel is opened earlier to allow the callee's unit to generate the ringback and send it to the caller.

URI-Parameters

You can specify whether or not the Mediatrix 4100 copies the *uri-parameters* from the Request-URI header to the level of proxy authentication. Not copying the *uri-parameters* allows to reduce the SIP packet size but it does not follow the recommendations of RFC 3261.

► **To copy the uri-parameters to the level of proxy authentication:**

1. In the *sipInteropMIB*, set the *sipInteropProxyAuthenticationUriParametersEnable* variable to **enable**.

Unsupported INFO Request

You can define the Mediatrix 4100's behaviour upon reception of an unknown type of INFO request.

► **To define the unsupported INFO request behaviour:**

1. In the *sipInteropMIB*, set the *sipInteropAckUnsupportedInfoRequests* variable with the proper behaviour.

Table 178: Unsupported INFO Request Behaviour

Parameter	Description
disable	Unknown INFO requests trigger a 415 Unsupported Media Type response.
enable	Reception of unknown INFO requests is acknowledged with a 200 OK response.

Outbound Proxy Usage

You can define whether or not SIP requests sent to the proxy contain a *Route* header displaying the proxy's address. It is only effective when an outbound proxy host is configured (see [“Outbound Proxy Server” on page 168](#) for more details). It is useful when interoperating with SIP servers that are not in conformance with RFC 3261's recommended practice concerning outbound proxy usage.

► **To set the outbound proxy usage:**

1. In the *sipInteropMIB*, set the *sipInteropRemoveOutboundProxyRouteHeader* variable with the proper behaviour.

Table 179: Outbound Proxy Usage

Parameter	Description
disable	SIP requests sent to the outbound proxy contain a <i>Route</i> header, as per RFC 3261's recommendation. This is the default behaviour.
enable	SIP requests are routed to the outbound proxy without inserting a <i>Route</i> header in the SIP packet.

International Code Mapping

Some international calling parties have their caller ID prepended with the “+” character. You can instruct the Mediatrix 4100 to substitute this “+” character with a user-defined value.

► **To define an international code mapping:**

1. In the *sipInteropMIB*, set the *sipInteropInternationalCodeMappingString* variable with the character string that is substituted to the “+” character that prepends some international caller IDs.
2. Set the *sipInteropInternationalCodeMappingEnable* to **enable**.
The default value is *disable*, which means that no substitution is performed and the '+' character is simply removed.

T.38 Negotiation Syntax

Standards Supported

- ITU-T Recommendation T.38, section D.2.3

You can define the format used, in the SDP portion of SIP packets, to advertise the unit's T.38 capabilities.

► **To set the T.38 negotiation syntax to use:**

1. In the *sipInteropMIB*, set the *sipInteropUseIuT38Format* variable with the proper behaviour.

Table 180: T.38 Negotiation Syntax Usage

Parameter	Description
disable	Support for the boolean T.38 parameters T38FaxFillBitRemoval, T38FaxTranscodingMMR, and T38FaxTranscodingJBIG is advertised by associating a value of 0 (unsupported) or 1 (supported) with the parameter in the following manner: <pre>a=T38FaxFillBitRemoval:0 a=T38FaxTranscodingMMR:0 a=T38FaxTranscodingJBIG:0</pre> This is the default value.
enable	Support for the above T.38 parameters is advertised in conformance with ITU-T Recommendation T.38, section D.2.3. The presence of the parameter in the SDP indicates support for it (without the need for an associated value), while its absence means that it is not supported.

Addressing Failed Registration Attempts

You can control whether or not failed registration attempts are retried periodically.

► **To address failed registration attempts:**

1. In the *sipInteropMIB*, set the *sipInteropRetryFailedRegistration* variable with the proper behaviour.

Table 181: Failed Registration Attempts Behaviour

Parameter	Description
disable	No retries are performed following a failed registration attempt. Manual intervention is required for the port to re-attempt registration.
enable	A failure to register a port to the SIP Registrar triggers an automatic retry every 2 minutes. This is the default value.

SIP Domain in Request URI

You can control whether or not the request URI is built using the SIP Domain.

► **To use the SIP domain in the Request URI:**

1. In the *sipInteropMIB*, set the *sipInteropUseSipDomainInRequestURI* variable with the proper behaviour.

Table 182: SIP Domain in Request URI Behaviour

Parameter	Description
disable	The request URI is built using the home domain proxy host as defined in “Proxy Server” on page 167 .
enable	The request URI is built using the SIP domain, if one is specified, for all SIP methods with the exception of REGISTER and PUBLISH. The parameter maddr is also added to the Request URI with the value of the home domain proxy host. See “User Agents” on page 259 for information on how to set the SIP domain.

SIP From: URI Content

You can specify the composition of the domain used in the from : URI of SIP requests sent by the unit.

► **To specify the composition of the domain used in the from : URI:**

1. In the *sipInteropMIB*, set the *sipInteropFromUriDomainSelection* variable with the proper behaviour.

Table 183: SIP From: URI Content

Parameter	Description
sipDomain	The domain of from:URI is built from the <i>sipDomain</i> variable (“User Agents” on page 259). If <i>sipDomain</i> is empty, the <i>sipHomeDomainProxyHost</i> value is used instead (“Proxy Server” on page 167). This is the default value.
localHostWanAddress	The domain of from:URI is built from the unit address taken from the <i>localHostWanAddress</i> variable (“WAN Address Configuration Source” on page 146).
localHostFqdn	The domain of from:URI is built from the unit's FQDN taken from the <i>localHostFqdn</i> variable. If <i>localHostFqdn</i> is empty, the value of the <i>localHostWanAddress</i> variable is used instead.

Network Asserted Caller ID

The Mediatrix 4100 can extract the caller ID information from the *P-Asserted-Identity* header (a SIP extension described in RFC 3325) instead of the *From* header of the incoming SIP request

► **To use the P-Asserted-Identity header:**

1. In the *sipInteropMIB*, set the *sipInteropUsePAssertedHeader* variable with the proper behaviour.

Table 184: Network Asserted Caller ID Behaviour

Parameter	Description
disable	Caller ID data is based on the value assigned to the <i>From</i> header of the incoming SIP request. This is the default value.
enable	The unit first attempts to get the caller ID information from the <i>P-Asserted-Identity</i> header, if present. In case of failure, it falls back to the content of the <i>From</i> header.

Payload Type Settings

The following are settings related to the DTMF payload type.

Using the Payload Type Found in the Answer

The default behaviour when sending an initial offer that contains an RFC 2833 payload type is to keep using that payload type even if the response comes back with a different one. You can set the Mediatrix 4100 to rather use the payload type found in the answer.

► **To use the payload type found in the answer:**

1. In the *sipInteropMIB*, set the *sipInteropUseDtmfPayloadTypeFoundInAnswer* variable with the proper behaviour.

Table 185: Payload Type in Answer

Parameter	Description
disable	Keep using the initial payload type. This is the default value.
enable	Use the RFC 2833 payload type found in the received answer

This variable only has an effect when the *voicelfDtmfTransport* variable is set to **outOfBandUsingRtp** (see [“DTMF Transport Type” on page 237](#) for more details). The payload type is used symmetrically meaning that it is used to send and receive the DTMF. Use the variable *sipInteropAllowAsymmetricDtmfPayloadType* to allow asymmetric payload type (see [“Asymmetric DTMF Payload Type” on page 282](#) for more details).

Asymmetric DTMF Payload Type

The default behaviour when receiving an answer to an offer that contained an RFC 2833 payload type is to use that payload type symmetrically (to send and receive DTMF). You can set the Mediatrix 4100 to rather use the payload type that was placed in the initial offer to receive DTMF but still use the one in the response to send them.

► To use the asymmetric DTMF payload type:

1. In the *sipInteropMIB*, set the *sipInteropAllowAsymmetricDtmfPayloadType* variable with the proper behaviour.

Table 186: Asymmetric DTMF Payload Type

Parameter	Description
disable	Use the RFC 2833 payload type found in the received answer to receive DTMF. This is the default value.
enable	Keep using the initial payload type to receive DTMF.

The variable does not affect the behaviour when receiving an offer. It only has an effect when the variable *sipInteropUseDtmfPayloadTypeFoundInAnswer* is set to **enable** (see [“Using the Payload Type Found in the Answer” on page 282](#) for more details).

Controlling the Call Waiting Tone via SIP INFO

The Mediatrix 4100 supports receiving some Call Waiting control commands via the SIP INFO method. Currently, the only supported content-type is “application/broadsoft”.

The controlled call waiting tone is played through the telephony interface.

► To control the call waiting tone via SIP INFO:

1. In the *sipInteropMIB*, set the *sipInteropCallWaitingToneControlViaSipInfo* variable with the proper behaviour.

Table 187: Call Waiting Tone Control

Parameter	Description
disable	The application rejects the SIP INFO with the content type 'application/broadsoft'. This is the default value.
enable	SIP INFO with the content type 'application/broadsoft' is accepted and the call waiting tone is started or stopped according to the body. The INFO method is also included in the 'Allow' SIP header.

Ignore Username Parameter

You can control whether or not the username parameter is ignored when routing an incoming SIP call to a line.

The default behaviour in the interpretation of the username parameter of an incoming request for the routing purpose is to compare the entire username. You can configure the Mediatrix 4100 so that it ignores the username parameter in the comparison.

The username parameters are all characters following a ';' in the username part of the URI.

► To ignore the username parameter:

1. In the *sipInteropMIB*, set the *sipInteropIgnoreUsernameParam* variable with the proper behaviour.

Table 188: Ignore Username Parameter

Parameter	Description
disable	The username parameter is not ignored when routing an incoming SIP call to a line. The parameter is considered as part of the username. This is the default value.
enable	The username parameter is ignored when routing an incoming SIP call to a line.

Escaping the Pound (#) Character in SIP URI Username

You can define whether or not the pound character (#) must be escaped in the username part of a SIP URI

► **To define whether or not the pound character must be escaped:**

1. In the *sipInteropMIB*, set the *sipInteropEscapePoundInSipUriUsername* variable with the proper behaviour.

Table 189: Escaping Pound Character Parameter

Parameter	Description
Enable	The Pound character (#) is escaped in the username part of a SIP URI.
Disable	The Pound character (#) is not escaped in the username part of a SIP URI. Note that RFC 3261 specifies that the pound character (#) needs to be escaped in the username part of a SIP URI.

SIP OPTIONS Method Support

You can define the behaviour of the Mediatix 4100 when answering a SIP OPTIONS request.

► **To define the SIP OPTIONS method support:**

1. In the *sipInteropMIB*, set the *sipInteropSipOptionsMethodSupport* variable with the proper behaviour.

Table 190: SIP OPTIONS Method Support Parameter

Parameter	Description
none	The Mediatix 4100 responds with an error 405 Method not allowed.
alwaysOk	The Mediatix 4100 responds with a 200 OK regardless of the content of the OPTIONS request.

Offer/Answer Model

Standards Supported

- RFC 3264 – An Offer/Answer Model with the Session Description Protocol (SDP)

You can define whether or not the Mediatix 4100 requires strict adherence to RFC 3264 from the peer when negotiating capabilities for the establishment of a media session.

► **To define how to process the Offer/Answer model:**

1. In the *sipExperimentalMIB*, set the *sipEnforceOfferAnswerModel* variable to the proper value.

Table 191: Offer/Answer Model Parameters

Parameter	Description
disable	The peer can freely: • Send back a brand new list of codecs or add new ones to the offered list. • Add media lines AFTER the ones found in the offer. As long as at least one codec sent back is supported by the Mediatix 4100, the call is allowed to go on. Any media line added by the peer is simply ignored.

Table 191: Offer/Answer Model Parameters

Parameter	Description
enable	The following guidelines from the Offer-Answer Model must be strictly followed. An answer must: • Include at least one codec from the list that the Mediatix 4100 sent in the offer. • Avoid adding extra codecs that were not present in the offer. • Contain the same number of media lines that the unit put in its offer. Otherwise, the answer is rejected and the unit ends the call. This is the default value.

The *sipAllowMediaReactivationInAnswerEnable* (see [“Allow Media Reactivation in Answer” on page 285](#)) and *sipAllowAudioAndImageNegotiationEnable* (see [“Allow Audio and Image Negotiation” on page 285](#)) variables allow to enable or disable other deviations from the Offer/Answer model.

Allow Media Reactivation in Answer

You can define the behaviour of the Mediatix 4100 when receiving a SDP answer activating a media that had been previously deactivated in the offer.

You can also set this parameter via the web interface, as described in [“Interop Parameters” on page 69](#).

► To define the behaviour when receiving a SDP answer:

1. In the *sipExperimentalMIB*, set the *sipAllowMediaReactivationInAnswerEnable* variable with the proper behaviour.

Table 192: Media Reactivation Parameters

Parameter	Description
Enable	A media reactivated in an incoming answer is ignored. This behaviour goes against the SDP Offer/Answer model described by IETF RFC 3264.
Disable	A media reactivated in an incoming answer ends the current media negotiation and the call. This behaviour follows the SDP Offer/Answer model described by IETF RFC 3264.

The *sipEnforceOfferAnswerModel* (see [“Offer/Answer Model” on page 284](#)) and *sipAllowAudioAndImageNegotiationEnable* (see [“Allow Audio and Image Negotiation” on page 285](#)) variables allow to enable or disable other deviations from the Offer/Answer model.

Allow Audio and Image Negotiation

You can define the behaviour of the Mediatix 4100 when offering media or answering to a media offer with audio and image negotiation.

You can also set this parameter via the web interface, as described in [“Interop Parameters” on page 69](#).

► **To allow audio and image negotiation:**

1. In the *sipExperimentalMIB*, set the *sipAllowAudioAndImageNegotiationEnable* variable with the proper behaviour.

Table 193: Audio and Image Negotiation Parameters

Parameter	Description
Enable	The unit offers audio and image media simultaneously in outgoing SDP offers and transits to T.38 mode upon reception of a T.38 packet. Also, when the unit answers positively to a SDP offer with audio and image, it transits to T.38 mode upon reception of a T.38 packet.
Disable	Outgoing offers never include image and audio simultaneously. Incoming offers with audio and image media with a non-zero port are considered as offering only audio.

The *sipEnforceOfferAnswerModel* (see [“Offer/Answer Model” on page 284](#)) and *sipAllowMediaReactivationInAnswerEnable* (see [“Allow Media Reactivation in Answer” on page 285](#)) variables allow to enable or disable other deviations from the Offer/Answer model.

Codec Order in Answer

You can define the behaviour of the Mediatix 4100 when answering to a media offer.

► **To define the codec order in answer:**

1. In the *sipExperimentalMIB*, set the *sipCodecOrderInAnswer* variable with the proper behaviour.

Table 194: Codec Order in Answer Parameters

Parameter	Description
localOrder	The codecs contained in the answer are prioritized according to the configured preferred codec. This means that the codecs in the answer may have a different order than in the received offer.
offerOrder	The codecs contained in the answer have the same order as the received offer. This means that the preferred codec configured has no effect on the codec order of the answer.

UDP Checksum

You can enable the UDP checksum in the RTP stack.

A checksum is a count of the number of bits in a transmission unit that is included with the unit so that the receiver can check to see whether or not the same number of bits arrived. If the counts match, it is assumed that the complete transmission was received.

UDP checksumming is disabled by default to gain performance. In this case, the UDP checksum is not calculated. The value 0 is written in the *checksum* field of each RTP packet header.

This feature is currently located under the *mediatrixExperimental* branch of the MIB structure. See [“MIB Structure” on page 135](#) for more details.

► **To enable UDP checksum:**

1. In the *sipExperimentalMIB*, set the *sipRtpUdpChecksumEnable* variable to **enable**.
The UDP checksum is calculated for each RTP packet. Note that this may affect performance. The parameter takes effect on the next connection.

This chapter describes how to configure the STUN client of the Mediatrix 4100.

What is STUN?

Standards Supported

RFC 3489 – STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs)

STUN (Simple Traversal of UDP through NATs) is a simple client / server protocol that uses UDP packets to discover the configuration information of NATs and firewalls between a device and the public Internet:

- ▶ NAT type
- ▶ NAT binding public address
- ▶ NAT binding time to live

NAT (Network Address Translator) is a device that translates the IP address used within a “private” network to a different IP address known in another “public” network. See [“NAT Traversal” on page 264](#) for more details.

STUN supports a variety of existing NAT devices and does not require any additional hardware or software upgrades on the NAT device.

The Mediatrix 4100 uses the STUN protocol to discover its NAT binding for the following three IP addresses/ports (sockets):

- ▶ Signalling protocol (SIP) IP address/port
- ▶ RTP IP address/port
- ▶ T.38 IP address/port

SIP Outbound Proxy

For a unit to work properly behind a firewall, it must keep a pinhole opened by sending keepalive packets through the firewall.

The Mediatrix 4100 only sends keepalive packets to the last destination for a specific socket. When a unit is not configured with an outbound proxy, it can send, through its SIP socket, messages to various destinations, such as a SIP redirect server, another SIP unit, or a MWI server. If, for instance, the last SIP message was sent to the MWI server, the Mediatrix 4100 will keep the pinhole opened for the MWI server only (sending keepalive message to the MWI server) and won't be reachable by other units outside the firewall.

To avoid those issues, all SIP message should come and go from the same source/destination on the public side of the firewall, i.e., a SIP outbound proxy. Media5 thus recommends that you use a SIP outbound proxy. See [“Outbound Proxy Server” on page 168](#) for more details.

Restrictions on the Media5 STUN Implementation

- ▶ The Mediatrix 4100 does not currently support NAT type discovery.
- ▶ The Mediatrix 4100 does not currently support STUN NAT binding time to live discovery.
- ▶ The Mediatrix 4100 does not currently support the TLS security mechanism.
- ▶ Due to a limitation of most routers, an RTP portal might be required in order for two units behind the same NAT/firewall to be able to communicate with each other.

STUN Client Configuration

The following describes how to configure the Mediatatrix 4100 STUN client via SNMP. You can also use the web interface to configure the STUN parameters. See [“STUN Configuration” on page 121](#) for more details.

► **To configure the STUN client:**

1. In the *ipAddressConfig* folder, locate the *ipAddressConfigStunStatic* group.
No DHCP value is available, you can only define STUN server information with static values.
2. Set the static STUN server IP address or Fully Qualified Domain Name (FQDN) in the *stunStaticHost* variable.
The default value is **192.168.0.10**.
3. Set the static STUN server IP port number in the *stunStaticPort* variable.
The default value is **3478**.
4. In the *stunMIB*, set the amount of time, in seconds, the Mediatatrix 4100 should keep a STUN query result in its internal cache in the *stunQueryCacheDuration* variable.
Keeping a query in a cache helps to reduce the amount of unnecessary STUN queries when an RTP or T.38 socket is re-used within a short period of time. Available values range from 0 s to 3600 s.
When set to **0**, the cache is disabled and the unit performs a STUN query each time a socket needs to be used.
5. Set the maximum amount of time, in milliseconds, the Mediatatrix 4100 should wait for an answer to a STUN query sent to a STUN server in the *stunQueryTimeout* variable.
Available values range from 500 ms to 10000 ms. The default value is 1000 ms.
Caution is advised in setting long timeouts. In the advent of an unresponsive STUN server, the unit may end up waiting a long time before it determines that a call cannot be made due to the STUN server failure.
6. Define the interval, in seconds, at which the Mediatatrix 4100 sends blank keepalive messages to keep a firewall hole opened in the *stunKeepAliveInterval* variable.
Keepalive messages are used by both the signalling protocol socket and the RTP socket to keep those connections opened through a firewall. Available values range from 0 s to 120 s. The default value is 30 s.
When set to **0**, no keepalive packet is sent.



Note: Keepalive messages are not supported on the T.38 socket.

7. Enable the STUN client by setting the *stunEnable* variable to **enable**.
This enables the STUN client for all sockets (VoIP signalling, RTP and T.38) altogether.
The following behaviour also applies:
 - If a unit is unable to re-register and there are no ongoing calls, it tries to rediscover its NAT binding for the signalling protocol socket.
 - If a STUN server is unresponsive, it is put in a “penalty box” for 60 seconds. See [“SIP Penalty Box” on page 267](#) for more details.
8. Restart the Mediatatrix 4100 so that the changes may take effect.

The Simple Network Time Protocol (SNTP) enables the notion of time (date, month, time) into the Mediatrix 4100. It updates the internal clock of the unit, which is the client of a SNTP server. It is required when dealing with features such as the caller ID.

SNTP is used to synchronize a SNTP client with a SNTP or NTP server by using UDP as transport.

You can also set these parameters via the web interface, as described in [“SNTP Settings” on page 37](#).

Enabling the SNTP Client

Standards Supported	RFC 1769 – Simple Network Time Protocol (SNTP)
---------------------	--

You must enable the SNTP client of the Mediatrix 4100 to properly connect to a a SNTP or NTP server.

► To enable the SNTP feature:

1. In the *sntpMIB*, set the *sntpEnable* variable to **enable**.
2. Set the following synchronization information:

Table 195: SNTP Synchronization Information

Variable	Description
sntpSynchronizationPeriod	Time interval (in minutes) between requests made to the SNTP server. The result is used to synchronize the unit with the time server. The maximum value is set to 1440 minutes (24 hours). Default Value: 1440
sntpSynchronizationPeriodOnError	Time interval (in minutes) between retries after an unsuccessful attempt to reach the SNTP server. The maximum value is set to 1440 (24 hours). Default Value: 60

Configuration Source

The Mediatrix 4100 must know the IP address and port number of the SNTP server. You can assign these information to the Mediatrix 4100 through a DHCP server or manually enter them yourself with the static variables.



In the *Unit Manager Network Administration Manual*, refer to chapter *Administration Parameters*, section *SNTP*.

DHCP Configuration

Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See [“Chapter 8 - IP Address and Network Configuration” on page 143](#) for more details.

► **To use DHCP-assigned information:**

1. In the *ipAddressConfig* folder, locate the *sntpSelectConfig Source* variable (under the *ipAddressConfigSntp* group).
This variable defines whether the Mediatrix 4100 must ask for its SNTP server settings through a DHCP server or not.
2. Set the *sntpSelectConfigSource* variable to **dhcp**.
You can query the SNTP server's IP address and port number assigned by the DHCP server in the *sntpHost* and *sntpPort* read-only variables (under the *ipAddressStatusSntp* group).
3. Set the DHCP Vendor Specific code of the SNTP feature in your DHCP server.
See [“SNTP” on page 152](#) for more details.

Static Configuration

Use the static configuration if you are not using a DHCP server or if you want to bypass it.

► **To use static information:**

1. In the *ipAddressConfig* folder, locate the *sntpSelectConfig Source* variable (under the *ipAddressConfigSntp* group).
This variable defines whether the Mediatrix 4100 must ask for its SNTP server settings through a DHCP server or not.
2. Set the *sntpSelectConfigSource* variable to **static**.
3. Set the following variables:

Table 196: SNTP Static Address

Variable	Description
sntpStaticHost	Static SNTP server IP address or domain name. Default Value: 192.168.0.10
sntpStaticPort	Static SNTP server IP port number. Default Value: 123

Defining a Custom Time Zone

Standards Supported

bootp-dhcp-option-88.txt Internet draft

When starting, the Mediatrix 4100 queries a NTP or SNTP server to receive time information. It receives the information in Greenwich Mean Time (GMT) format (also known as Universal Time Coordinated - UTC), so it needs to convert this GMT time into the proper time zone. To do this, the Mediatrix 4100 offers time zone configuration with daylight saving settings.

► To define a custom time zone:

1. In the *sntpMIB*, enter a valid POSIX (Portable Operating System Interface) string in the *sntpTimeZoneString* variable as defined in the <bootp-dhcp-option-88.txt> Internet draft.
The format of the string is validated upon entry. Invalid entries are refused. The default value is:
EST5DST4,M4.1.0/02:00:00,M10.5.0/02:00:00
A POSIX string is a set of standard operating system interfaces based on the UNIX operating system. The format of the IEEE 1003.1 POSIX string is defined in the <bootp-dhcp-option-88.txt> Internet draft as:
STD0FFSET[DST[OFFSET],[START[/TIME],END[/TIME]]]
Refer to the following sub-sections for explanations on each part of the string.

STD / DST

Three or more characters for the standard (STD) or alternative daylight saving time (DST) time zone. Only STD is mandatory. If DST is not supplied, the daylight saving time does not apply. Lower and upper case letters are allowed. All characters are allowed except digits, leading colon (:), comma (,), minus (-), plus (+), and ASCII NUL.

OFFSET

Difference between the GMT time and the local time. The offset has the format *h[h][:m[m]][:s[s]]*. If no offset is supplied for DST, the alternative time is assumed to be one hour ahead of standard time. One or more digits can be used; the value is always interpreted as a decimal number.

The hour value must be between 0 and 24. The minutes and seconds values, if present, must be between 0 and 59. If preceded by a minus sign (-), the time zone is east of the prime meridian, otherwise it is west, which can be indicated by the preceding plus sign (+). For example, New York time is GMT 5.

START / END

Indicates when to change to and return from the daylight saving time. The *START* argument is the date when the change from the standard to the daylight save time occurs; *END* is the date for changing back. If *START* and *END* are not specified, the default is the US Daylight saving time start and end dates. The format for start and end must be **one** of the following:

- **n** where *n* is the number of days since the start of the year from 0 to 365. It must contain the leap year day if the current year is a leap year. With this format, you are responsible to determine all the leap year details.
- **Jn** where *n* is the Julian day number of the year from 1 to 365. Leap days are not counted. That is, in all years – including leap years – February 28 is day 59 and March 1 is day 60. It is impossible to refer to the occasional February 29 explicitly. The *TIME* parameter has the same format as *OFFSET* but there can be no leading minus (-) or plus (+) sign. If *TIME* is not specified, the default is 02:00:00.
- **Mx[x].y.z** where *x* is the month, *y* is a week count (in which the *z* day exists) and *z* is the day of the week starting at 0 (Sunday). As an example:
M10.4.0
is the fourth Sunday of October. It does not matter if the Sunday is in the 4th or 5th week.
M10.5.0

is the last Sunday of October (5 indicates the last z day). It does not matter if the Sunday is in the 4th or 5th week.

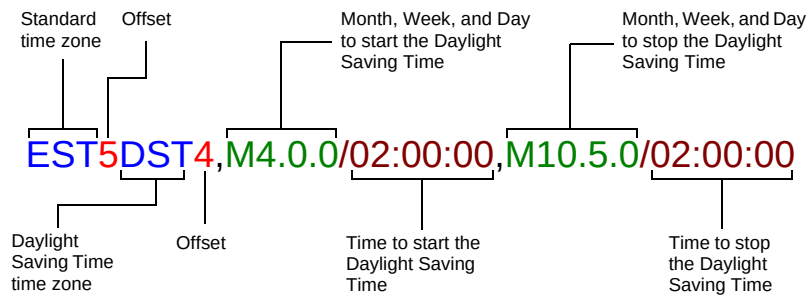
M10.1.6

is the first week with a Saturday (thus the first Saturday). It does not matter if the Saturday is in the first or second week.

The *TIME* parameter has the same format as *OFFSET* but there can be no leading minus (-) or plus (+) sign. If *TIME* is not specified, the default is 02:00:00.

Example

The following is an example of a proper POSIX string:



The following are some valid POSIX strings:

Table 197: Valid POSIX Strings

Time Zone	POSIX String
Pacific Time (Canada & US)	PST8PDT7,M3.2.0/02:00:00,M11.1.0/02:00:00
Mountain Time (Canada & US)	MST7MDT6,M3.2.0/02:00:00,M11.1.0/02:00:00
Central Time (Canada & US)	CST6CDT5,M3.2.0/02:00:00,M11.1.0/02:00:00
Eastern Time Canada & US)	EST5EDT4,M3.2.0/02:00:00,M11.1.0/02:00:00
Atlantic Time (Canada)	AST4ADT3,M3.2.0/02:00:00,M11.1.0/02:00:00
GMT Standard Time	GMT0DMT-1,M3.5.0/01:00:00,M10.5.0/02:00:00
W. Europe Standard Time	WEST-1DWEST-2,M3.5.0/02:00:00,M10.5.0/03:00:00
China Standard Time	CST-8
Tokyo Standard Time	TST-9
Central Australia Standard Time	CAUST-9:30DCAUST-10:30,M10.5.0/02:00:00,M3.5.0/02:00:00
Australia Eastern Standard Time	AUSEST-10AUSDST-11,M10.5.0/02:00:00,M3.5.0/02:00:00
UTC (Coordinated Universal Time)	UTC0

This chapter describes how to use a digit map.

Standards Supported

RFC 2705 – Media Gateway Control Protocol (MGCP) Version 1.0, section 3.4 (Formal syntax description of the protocol).



In the *Unit Manager Network Administration Manual*, refer to chapter *Dial Map Parameters*.

You can also set these parameters via the web interface, as described in [“Digit Maps” on page 73](#).

What is a Digit Map?

A digit map allows you to compare the number users just dialed to a string of arguments. If they match, users can make the call. If not, users cannot make the call and get an error signal. It is thus essential to define very precisely a digit map before actually implementing it, or your users may encounter calling problems.

Because the Mediatrix 4100 cannot predict how many digits it needs to accumulate before transmission, you could use the digit map, for instance, to determine exactly when there are enough digits entered from the user to place a call.

Syntax

The permitted digit map syntax is taken from the core MGCP specification, RFC 2705, section 3.4:

```
DigitMap = DigitString / '(' DigitStringList ')'
DigitStringList = DigitString 0* ( '|' DigitString )
DigitString = 1*(DigitStringElement)
DigitStringElement = DigitPosition ['.']
DigitPosition = DigitMapLetter / DigitMapRange
DigitMapLetter = DIGIT / '#' / '*' / 'A' / 'B' / 'C' / 'D' / 'T'
DigitMapRange = 'x' / '[' 1*DigitLetter ']'
DigitLetter ::= *((DIGIT '-' DIGIT) / DigitMapLetter)
```

Where “x” means “any digit” and “.” means “any number of”.

For instance, using the telephone on your desk, you can dial the following numbers:

Table 198: Number Examples

Number	Description
0	Local operator
00	Long distance operator
xxxx	Local extension number
8xxxxxxx	Local number
#xxxxxxx	Shortcut to local number at other corporate sites
91xxxxxxxxxx	Long distance numbers
9011 + up to 15 digits	International number

The solution to this problem is to load the Mediatix 4100 with a digit map that corresponds to the dial plan. A Mediatix 4100 that detects digits or timers applies the current dial string to the digit map, attempting a match to each regular expression in the digit map in lexical order.

- ▶ If the result is under-qualified (partially matches at least one entry in the digit map), waits for more digits.
- ▶ If the result matches, dials the number.
- ▶ If the result is over-qualified (i.e., no further digits could possibly produce a match), sends a fast busy signal.

Special Characters

Digit maps use specific characters and digits in a particular syntax. Those characters are:

Table 199: Digit Map Characters

Character	Use
Digits (0, 1, 2... 9)	Indicates specific digits in a telephone number expression.
T	The Timer indicates that if users have not dialed a digit for the time defined, it is likely that they have finished dialing and the SIP Server can make the call.
x	Matches any digit, excluding “#” and “*”.
	Indicates a choice of matching expressions (OR).
.	Matches an arbitrary number of occurrences of the preceding digit, including 0.
[	Indicates the start of a range of characters.
]	Indicates the end of a range of characters.

How to Use a Digit Map

Let's say you are in an office and you want to call a co-worker's 3-digits extension. You could build a digit map that says “after the user has entered 3 digits, make the call”. The digit map could look as follows:

xxx

You could refine this digit map by including a range of digits. For instance, you know that all extensions in your company either begin with 2, 3, or 4. The corresponding digit map could look as follows:

[2-4]xx

If the number you dial begins with anything other than 2, 3, or 4, the call is not placed and you get a busy signal.

Combining Several Expressions

You can combine two or more expressions in the same digit map by using the “|” operator, which is equal to OR.

Let's say you want to specify a choice: the digit map is to check if the number is internal (extension), or external (a local call). Assuming that you must first dial “9” to make an external call, you could define a digit map as follows:

([2-4]xx|9[2-9]xxxxxx)

The digit map checks if:

- ▶ the number begins with 2, 3, or 4 **and**
- ▶ the number has 3 digits

If not, it checks if:

- ▶ the number begins with 9 **and**
- ▶ the second digit is any digit between 2 and 9 **and**
- ▶ the number has 7 digits



Note: Enclose the digit map in parenthesis when using the “[]” option.

Using the # and * Characters

It may sometimes be required that users dial the “#” or “*” to make calls. This can be easily incorporated in a digit map:

```
xxxxxxx#
xxxxxxx*
```

The “#” or “*” character could indicate users must dial the “#” or “*” character at the end of their number to indicate it is complete. You can specify to remove the “#” or “*” found at the end of a dialed number. See [“Setting up Digit Maps” on page 296](#).

Using the Timer

You can configure the Timer. See [“Digit Maps Timeouts” on page 298](#) for more details. It indicates that if users have not dialed a digit for the time defined, it is likely that they have finished dialing and the Mediatrix 4100 can make the call. A digit map for this could be:

```
[2-9]xxxxxxT
```



Note: When making the actual call and dialing the number, the Mediatrix 4100 automatically removes the “T” found at the end of a dialed number, if there is one (after a match). This character is for indication purposes only.

Calls Outside the Country

If your users are making calls outside their country, it may sometimes be hard to determine exactly the number of digits they must enter. You could devise a digit map that takes this problem into account:

```
001x.T
```

In this example, the digit map looks for a number that begins with 001, and then any number of digits after that (x.).

Example

[Table 198 on page 293](#) outlined various call types one could make. All these possibilities could be covered in one digit map:

```
(0T|00T|[1-7]xxx|8xxxxxxx|#xxxxxxx|91xxxxxxxxxxx|9011x.T)
```

Validating a Digit Map

The Mediatrix 4100 validates the digit map as you are entering it and it forbids any invalid value.

Processing Digits When Pressed

You can define whether or not a call is made once all DTMFs have been verified against a DTMF map.

► **To define how to process digit maps:**

1. In the *digitMapMIB*, define when a digit is processed through the digit maps in the *digitMapProcessDigitsWhenPressed* variable.

Table 200: Processing Digits Parameters

Parameter	Description
disable	Digits are processed only when released. Disabling this feature increases the delay needed to match a dialed string to a digit map. There is also an impact on the <i>digitMapTimeoutFirstDigit</i> and <i>digitMapTimeoutCompletion</i> parameters (“Digit Maps Timeouts” on page 298) since the timers are stopped at the end of a DTMF instead of the beginning. It can also lead to small DTMF leaks when using subscriber services if the variable <i>subscriberServicesProcessingTrigger</i> is set to flashHookAndDigits (“Service Activation Processing” on page 305).
enable	Digits are processed as soon as they are pressed. This can lead to a DTMF leak in the RTP at the beginning of a call if the voice stream is established before the last DTMF is released. This is the default value.

Setting up Digit Maps

The variables related to the digit maps are located in tables. You can create/edit ten digit maps for each Mediatrix 4100. Before changing a parameter value, build its corresponding table with your MIB browser's table functionality. Depending on the MIB browser you are using, the tables may not appear the same way. Digit map rules are checked sequentially. If a telephone number potentially matches two of the rules, the first rule encountered is applied.

Each of these digit map rules has six specific variables to define for the digit map to work properly.

► **To set up digit maps:**

1. In the *digitMapMIB*, define the digit map string that is considered valid when dialed in the *digitMapAllowedDigitMap* variable.

The string must use the syntax described in [“Digit Maps” on page 293](#). The string format is validated upon entry. Invalid entries are refused. The default value is **x.T**. A digit map string may have a maximum of 64 characters.
2. Define the amount of digits to remove from the beginning of the dialed number, after dialing, but before initiating the call, in the *digitMapPrefixedDigitRemovalCount* variable.

For instance, when dialing “1-819-xxx-xxxx”, specifying a value of “4” means that the call is started by using the number “xxx-xxxx”. The default value is **0**.

This rule is applied BEFORE applying both *digitMapSuffixStringToRemove* (Step 3) and *digitMapPrependedString* (Step 4).
3. Define the string to look for and remove, from the end of the dialed number, in the *digitMapSuffixStringToRemove* variable.

This is helpful if one of the digit maps contains a terminating character that must not be dialed.

For instance, let's take a digit map such as “25#”, in which the “#” signals that the user has finished entering digits. If you want to remove the “#”, specify “#” in this variable and the resulting number is “25”.

This rule is applied AFTER applying *digitMapPrefixedDigitRemovalCount* (Step 2) but BEFORE applying *digitMapPrependedString* (Step 4).

4. Define the string to insert at the beginning of the dialed number before initiating the call in the *digitMapPrependedString* variable.

For instance, let's say that you need to dial a special digit, "9", for all local calls. Dialing "xxx-xxxx" with a value of "9" would yield "9-xxx-xxxx" as the number with which to initiate the call.

This rule is applied AFTER applying both *digitMapPrefixedDigitRemovalCount* (Step 2) and *digitMapSuffixStringToRemove* (Step 3).

5. Specify the line(s) on which to apply the digit map in the *digitMapAllowedLineToApply* variable. The string has the following syntax:

- **all**: Applies to all lines.
- **,**: Separator between non-consecutive lists of lines or single line.
- **n**: A single line, where n is the line number.
- **m-n**: List of lines where m is the start line number and n is the end line number.



Note: Line duplication is not allowed. Lines must be specified in low to high order.

Example:

'1,3-6': Applies to lines 1, 3, 4, 5, and 6.

The default value is **all**.

6. Enable the digit map by setting the *digitMapAllowedEnable* variable to **enable**. When enabled, this digit map is recognised and accepted only if it is also valid.

Refused Digit Maps

A refused digit map forbids your users to call specific numbers; for instance, you want to accept all 1-8xx numbers except 1-801. You can create/edit ten refused digit maps for each Mediatix 4100.

► To set up refused digit maps:

1. In the *digitMapMIB*, define the digit map string that is considered invalid when dialed in the *digitMapRefusedDigitMap* variable.
The string must use the syntax described in ["Digit Maps" on page 293](#). The string format is validated upon entry. Invalid entries are refused. A digit map string may have a maximum of 64 characters.
2. Specify the line(s) on which to apply the digit map in the *digitMapRefusedLineToApply* variable. The string has the following syntax:
 - **all**: Applies to all lines.
 - **,**: Separator between non-consecutive lists of lines or single line.
 - **n**: A single line, where n is the line number.
 - **m-n**: List of lines where m is the start line number and n is the end line number.



Note: Line duplication is not allowed. Lines must be specified in low to high order.

Example:

'1,3-6': Applies to lines 1, 3, 4, 5, and 6.

The default value is **all**.

3. Enable the refused digit map by setting the *digitMapRefusedEnable* variable to **enable**. When enabled, this digit map is recognised and refused only if it is also valid.

Digit Maps Timeouts

You can define timeouts that apply to the whole unit when dialing a digit map.

► **To configure digit map timeouts:**

1. In the *digitMapMIB* (*digitMapTimeouts* group), define the total time the user has to dial the DTMF sequence in the *digitMapTimeoutCompletion* variable.
The timer starts when the dial tone is played. When the timer expires, the receiver off-hook tone is played.
This value is expressed in milliseconds (ms). Values range from 1000 ms to 180000 ms. The default value is **60000** ms.
2. Define the time between the start of the dial tone and the receiver off-hook tone, if no DTMF is detected, in the *digitMapTimeoutFirstDigit* variable.
This value is expressed in milliseconds (ms). Values range from 1000 ms to 180000 ms. The default value is **20000** ms.
3. Define the value of the "T" digit in the *digitMapTimeoutInterDigit* variable.
The "T" digit is used to express a time lapse between the detection of two DTMFs.
This value is expressed in milliseconds (ms). Values range from 500 ms to 10000 ms. The default value is **4000** ms.

Digit Map Examples

Digit Map Example 1 – Standard Calls

Let's say you are located in Seattle, Washington and you want to define digit map rules for your users. You must consider at least four possibilities:

- You are making a long distance call outside the country.
- You are making a long distance call outside your area code.
- You are making a local call outside your area code (in the 425 area code).
- You are making a local call in the same area code.

Digit Map Rule #1

This digit map rule checks for calls outside the country.

Table 201: Digit Map Rules #1 Settings

Variable	Setting
digitMapAllowedEnable	Enable
digitMapAllowedDigitMap	(011x.# 001x.T)
digitMapPrefixedDigitRemovalCount	3 A valid telephone number must contain a country code, an area code, and a number – the "011" part is not required.

Digit Map Rule #2

This digit map rule checks for long distance calls outside your area code.

Table 202: Digit Map Rules #2 Settings

Variable	Setting
digitMapAllowedEnable	Enable
digitMapAllowedDigitMap	1xxxxxxxxx
digitMapPrefixedDigitRemovalCount	1 The first digit “1” in the digit map indicates a user wants to call outside his or her own area code. It must be removed because it does not need to be expressed to the SIP Server. The SIP Server needs only to know the complete number of the called party (CC+AC+directory number).
digitMapPrependedString	1 (country code) A valid telephone number must contain a country code, an area code, and a number. The country code is missing in this number and must be added. Note that in this scenario, the country code is the same as the code used when the user wants to indicate a communication outside of his or her own area code. It is still good practice to have this number removed and to add the country code, even if these two numbers are the same.

Digit Map Rule #3

This digit map rule checks for local calls outside your area code (in the 425 Area Code).

Table 203: Digit Map Rules #3 Settings

Variable	Setting
digitMapAllowedEnable	Enable
digitMapAllowedDigitMap	425xxxxxxx
digitMapPrependedString	1 (country code) A valid telephone number must contain a country code, an area code, and a number. The country code is missing in this number and must be added.

Digit Map Rule #4

This digit map rule checks for local calls in the same area code.

Table 204: Digit Map Rules #4 Settings

Variable	Setting
digitMapAllowedEnable	Enable
digitMapAllowedDigitMap	([235-9]xxxxxx 45[1-9]xxxx 4[0-469]xxxxx)
digitMapPrependedString	1206 (country code and area code) A valid telephone number must contain a country code, an area code, and a number. The country code and area code are missing in this number and must be added.

Digit Map Example 2 – PBX Emulation

Let's say you are located in the 819 area code. You are in an office where you dial:

- ▶ 3 numbers to call one of your co-workers.
- ▶ "9" to get an external line.

The following four possibilities are considered:

- ▶ You are making an internal call to one of your co-workers.
- ▶ You are making a long distance call outside the country.
- ▶ You are making a long distance call outside your area code.
- ▶ You are making a local call in the same area code.

Digit Map Rule #1

This digit map rule checks for internal calls.

Table 205: Digit Map Rules #1 Settings

Variable	Setting
digitMapAllowedEnable	Enable
digitMapAllowedDigitMap	[1-8]xx

Digit Map Rule #2

This digit map rule checks for calls outside the country.

Table 206: Digit Map Rules #2 Settings

Variable	Setting
digitMapAllowedEnable	Enable
digitMapAllowedDigitMap	(9011x.# 9011x.T)
digitMapPrefixedDigitRemovalCount	4 A valid telephone number must contain a country code, an area code, and a number – the "9011" part is not required.

Digit Map Rule #3

This digit map rule checks for long distance calls outside your area code.

Table 207: Digit Map Rules #3 Settings

Variable	Setting
digitMapAllowedEnable	Enable
digitMapAllowedDigitMap	91xxxxxxxxxx
digitMapPrefixedDigitRemovalCount	2 The first digit "9" in the digit map indicates a user wants to make an external call, while the second digit "1" indicates a user wants to call outside his or her own area code (in North America). The two digits must be removed because they do not need to be expressed to the SIP Server. The SIP Server needs only to know the complete number of the called party (CC+AC+directory number).

Table 207: Digit Map Rules #3 Settings (Continued)

Variable	Setting
digitMapPrependedString	1 (country code) A valid telephone number must contain a country code, an area code, and a number. The country code is missing in this number and must be added. Note that in this scenario, the country code is the same as the code used when the user wants to indicate a communication outside of his or her own area code. It is still good practice to have this number removed and to add the country code, even if these two numbers are the same.

Digit Map Rule #4

This digit map rule checks for local calls in the same area code.

Table 208: Digit Map Rules #4 Settings

Variable	Setting
digitMapAllowedEnable	Enable
digitMapAllowedDigitMap	9[2-8]xxxxxx
digitMapPrefixedDigit RemovalCount	1 The first digit "9" in the digit map indicates a user wants to make an external call. It has to be removed because it does not need to be expressed to the SIP Server. The SIP Server needs only to know the complete number of the called party (CC+AC+directory number).
digitMapPrependedString	1819 (country code and area code) A valid telephone number must contain a country code, an area code, and a number. The country code and area code are missing in this number and must be added.

This chapter explains how to set the telephony variables of the Mediatix 4100 to define the way the unit handles calls.

Making Calls

Users with telephones or faxes connected to a Mediatix 4100 dial as if they were on a standard telephony system.

Complete Dialing Sequence

There are three ways to indicate the dialed number sequence is complete and the Mediatix 4100 can dial the number:

- ▶ The administrator has set up the dialing process so that you must end the telephone number with a particular character to indicate it is complete, e.g., a “#”.
- ▶ The administrator has set up the dialing process with a timer. This timer checks the dialing process and, when no further digits have been dialed for the time set by the administrator, it assumes the number is complete and dials it.
- ▶ The administrator has set up the Mediatix 4100 so it knows exactly how many digits it must collect before it places the call. It finds the number of digits to collect by looking at the first few numbers dialed. For example: a telephone number beginning by 1 should be followed by 10 more digits in North America.

Dialing a Telephone Number or Numerical Alias

This section assumes that the Mediatix 4100 is configured to do SCN emulation. The Mediatix 4100 could be configured to do any other kind of emulation, thus its users would simply have to dial as if they were using their old system.

▶ **To dial a Standard Call:**

1. Dial the telephone number as if you were using a standard telephone, with country code and area code when required.

Examples:

223
8298749
15145701234

A Standard Call uses the server to contact the remote dialed user. The server takes the decision about redirecting the call on the SCN or keeping it on the network. Keeping the call on the network takes precedence over redirecting it on the SCN. If the call needs to go on the SCN, the server redirects it to a proper analog gateway (such as the Mediatix 1204) that will place the call to the SCN network.



Note: You can dial one star numbers *xx (such as *69). These numbers are automatically inserted in the Request-URL of the SIP INVITE request.

► **To dial a Forced SCN call:**

1. Dial "***".
2. Dial the telephone number as if you were using a standard telephone, with country code and area code when required.

Examples:

**8298749

**15145701234

A Forced SCN Call allows you to specify that the user you want to reach is located on the SCN network. This leaves no decision to the server; it must find a proper gateway and place the call on the SCN. This option can be useful only when a SCN number is shadowed by a network number.



Note: A forced SCN call is only possible if an analog gateway such as the Mediatix 1204 is available on the IP network.

Emergency Call

The Emergency Call service (also called urgent gateway) allows a "911"-style service. It allows a user to dial a special digit map resulting in a message being sent to a specified urgent gateway, bypassing any other intermediaries.

If enabled, whenever the user dials the specified digit map, a message is sent to the target address.

You can also set these parameters via the web interface, as described in ["Emergency Call Configuration" on page 120](#).

► **To enable the emergency call service:**

1. In the *emergencyCallMIB*, locate the *emergencyCallUrgentGatewayEnable* variable (under the *emergencyCallUrgentGatewayCustomization* group).
This variable sets the usage state of the urgent gateway. Urgent messages bypass the outbound proxy and go directly to the urgent gateway.
2. Define the digits that users must dial to start the urgent gateway call feature in the *emergencyCallUrgentGatewayDigitMap* variable.
For instance, you could decide to put "*60" as the sequence a user must dial to start the urgent gateway service. This sequence must follow the syntax for digit maps (see ["Chapter 21 - Digit Maps" on page 293](#)). Dialing this digit map does not have any effect unless the service's status is "enabled".
The activating sequence is set for all the lines of the Mediatix 4100. You cannot have different sequences for each line.
3. Set the number to reach for an urgent call in the *emergencyCallUrgentGatewayTargetAddress* variable.
Accepted formats are:
 - telephone numbers (5551111)
 - SIP URLs such as "scheme:user@host". For instance, "sip:user@foo.com".

Note that this string is used literally, so cosmetic symbols (such as the dash in "555-xxxx") should not be present.

The Mediatrix 4100 offers subscriber services users can directly access on their telephone. However, you must set these services before they can be used.

Most of the variables related to the subscriber services are located in tables. These tables display the information for all lines. Before changing a parameter value, build its corresponding table with your MIB browser's table functionality.



In the *Unit Manager Network Administration Manual*, refer to chapter *Subscriber Services Parameters*, section *SIP Configuration Window*.

You can also set these parameters via the web interface, as described in [“Call Forward” on page 90](#) and [“Services” on page 96](#).

Service Activation Processing

The user can activate a service in two ways:

- ▶ By performing a standard flash hook.
- ▶ By performing a flash hook and entering a digit to activate a specific service. The digit dialed has a different behaviour depending on the current call context.

You can define which of these two methods is available to your users.

▶ **To define the service activation processing:**

1. In the *subscriberServicesMIB*, locate the *subscriberServicesProcessingTrigger* variable.
2. Select which action the user must perform to trigger a service change by setting the *subscriberServicesProcessingTrigger* variable:

Table 209: Service Activation Parameters

Parameter	Description
flashHook	The user must perform a flash hook to activate a service.

Table 209: Service Activation Parameters (Continued)

Parameter	Description
flashHookAndDigits	The user must perform a flash hook and enter a specific digit to activate a service. The digit dialed has a different behaviour depending on the current call context: - One call active and one waiting call: Flash hook then dial the digit 2: Answer the waiting call. - One call active and one call on hold: Flash hook then dial the digit 1: Terminate the active call and recover the call on hold. Flash hook then dial the digit 2: Hold the active call and recover the call on hold. Flash hook then dial the digit 3: Enter the conference mode. Flash hook then dial the digit 4: Transfer the call on hold to the active call. When hanging up in this context, the telephone rings to notify the user there is still a call on hold. - In conference mode: Flash hook then dial the digit 2: Return to one active call and one call on hold. Flash hook then dial the digit 4: Transfer the second active call to the first active call. When hanging up in this context, all calls are finished.

As example, the following are the steps to perform a conference call:

1. Call the first attendee.
2. Flash hook to put the first attendee on hold.
3. Call the second attendee.
The context is now one call active and one call on hold.
4. Flash hook then dial the digit 3 to start the conference call.

Call Hold

The Call Hold service allows the user to temporarily put an existing call on hold, usually by using the “flash” button of the telephone. The user can resume the call in the same way.

You must enable this service for the following services to work properly:

- ▶ Call Waiting
- ▶ Second Call
- ▶ Blind Transfer
- ▶ Attended Transfer
- ▶ Conference

Enabling Call Hold

You must enable this service before your users can use it.

► **To enable the call hold service:**

1. In the *subscriberServicesMIB*, locate the *subscriberServicesIfEnablingTable* group.
2. Set the *subscriberServicesHoldEnable* variable to **enable**.
Because this variable is located in a table, you can enable/disable the service on a per-line basis. You can find the current status of the service in the *subscriberServicesHoldStatus* read-only variable (under the *subscriberServicesIfStatusTable*).

Using Call Hold

The following is the procedure to use this service on the user's telephone.

► **To put the current call on hold:**

1. Perform a Flash-Hook by pressing the "Flash" button on your analog telephone.
This puts the call on hold. You can resume the call in the same way.

Second Call

The Second Call service allows a user with an active call to put the call on hold, and then initiate a new call on a second line. This service is most useful with the transfer and conference services.

The call hold service must be enabled for this service to work. See ["Call Hold" on page 306](#).

Enabling Second Call

You must enable this service before your users can use it.

► **To enable the second call service:**

1. In the *subscriberServicesMIB*, locate the *subscriberServicesIfEnablingTable* group.
2. Set the *subscriberServicesSecondCallEnable* variable to **enable**.
Because this variable is located in a table, you can enable/disable the service on a per-line basis. You can find the current status of the service in the *subscriberServicesSecondCallStatus* read-only variable (under the *subscriberServicesIfStatusTable*).

Using Second Call

The following is the procedure to use this service on the user's telephone.

► **To use the second call service:**

1. Perform a Flash-Hook by pressing the "Flash" button on your analog telephone.
This puts the call on hold and the second line is automatically connected to your line.
2. Initiate the second call.

Call Forward

The Call Forward service offers various ways to forward calls:

- ▶ Unconditional
- ▶ On Busy
- ▶ On No Answer

Unconditional

The Call Forward Unconditional feature allows users to forward all of their calls to another extension or line.

Setting up Call Forward Unconditional

You must configure and enable this service before your users can use it.

▶ **To set the Call Forward Unconditional feature:**

1. In the *subscriberServicesMIB*, locate the *subscriberServicesIfCallForwardActivationTable* group.
2. Set the status of the service in the *subscriberServicesCallForwardUnconditionalActivation* variable to **inactive** or **active**.
This variable starts the service (active) or stops the service (inactive).
If you want to let the user start or stop the service with his or her handset, you must enter a sequence of digits in steps 3 and 4. The *subscriberServicesCallForwardUnconditionalActivation* variable is automatically updated to reflect the activation status according to the user's setting.
3. Define the digits that users must dial to start the service in the *subscriberServicesCallForwardUnconditionalEnableDigitMap* variable (under the *subscriberServicesActivationDigitmaps* group).
Define this variable only if you want to let the user start the service with his or her handset. If you rather want to have the control over the activation/deactivation of the service, see Step 2.
For instance, you could decide to put “*70” as the sequence to activate the service. This sequence must be unique and follow the syntax for digit maps (see [“Chapter 21 - Digit Maps” on page 293](#)). Dialing this digit map does not have any effect unless the service's status is “enabled”.
The activating sequence is set for all the lines of the Mediatrix 4100. You cannot have a different sequence for each line.
4. Define the digits that users must dial to stop the service in the *subscriberServicesCallForwardUnconditionalDisableDigitMap* variable (under the *subscriberServicesActivationDigitmaps* group).
Define this variable only if you want to let the user stop the service with his or her handset. If you rather want to have the control over the activation/deactivation of the service, see Step 2.
For instance, you could decide to put “*71” as the sequence to deactivate the service. This sequence must be unique and follow the syntax for digit maps (see [“Chapter 21 - Digit Maps” on page 293](#)). Dialing this digit map does not have any effect unless the service's status is “enabled”.
The deactivating sequence is set for all the lines of the Mediatrix 4100. You cannot have a different sequence for each line.
5. Define the address to which forward incoming calls in the *subscriberServicesCallForwardUnconditionalForwardingAddress* variable.
Accepted formats are:
 - telephone numbers (5551111)
 - SIP URLs such as “scheme:user@host”. For instance, “sip:user@foo.com”.

This string is used literally, so cosmetic symbols (such as the dash in “555-xxxx”) should not be present.

Because this variable is located in a table, you can have a different string for each line.

6. Enable the Call Forward Unconditional by setting the *subscriberServicesCallForwardUnconditionalEnable* variable to **enable** (under the *subscriberServicesIfEnablingTable* group).

If you set the variable to **disable**, this will not disable the call forward, but will prevent the user from activating or deactivating the service. The user will not be able to use the digits used to start and stop the service.

Because this variable is located in a table, you can enable/disable the service on a per-line basis.

Using Call Forward Unconditional

When forwarding calls outside the system, a brief ring is heard on the telephone to remind the user that the call forward service is active. The user can still make calls from the telephone.

► To forward calls:

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence the system administrator has implemented to activate the call forward unconditional service.
This sequence could be something like *70.
4. Wait for the transfer tone (three “beeps”) followed by the dial tone.
5. Dial the number to which you want to forward your calls. Dial any access code if required.
6. Wait for three “beeps” followed by a silent pause.
The call forward is established.
7. Hang up your telephone.
The calls are checked against the digit maps set up by the system administrator.

► To check if the call forward has been properly established:

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial your extension or telephone number.
The call is forwarded to the desired telephone number.
4. Hang up your telephone.

► To cancel the call forward:

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence the system administrator has implemented to deactivate the call forward – unconditional service.
This sequence could be something like *71.
4. Wait for three “beeps” followed by a silent pause.
The call forward is cancelled.
5. Hang up your telephone.

On Busy

You can automatically forward the incoming calls of your users to a pre-determined target if they are already on the line. The user does not have any feedback that a call was forwarded.

Setting up Call Forward On Busy

You must configure and enable this service before your users can use it.

► To set the Call Forward On Busy feature:

1. In the *subscriberServicesMIB*, locate the *subscriberServicesIfCallForwardActivationTable* group.
2. Set the status of the service in the *subscriberServicesCallForwardOnBusyActivation* variable to **inactive** or **active**.
This variable starts the service (active) or stops the service (inactive).
If you want to let the user start or stop the service with his or her handset, you must enter a sequence of digits in steps 3 and 4. The *subscriberServicesCallForwardOnBusyActivation* variable is automatically updated to reflect the activation status according to the user's setting.
3. Define the digits that users must dial to start the service in the *subscriberServicesCallForwardOnBusyEnableDigitMap* variable (under the *subscriberServicesActivationDigitmaps* group).
Define this variable only if you want to let the user start the service with his or her handset. If you rather want to have the control over the activation/deactivation of the service, see Step 2.
For instance, you could decide to put “*72” as the sequence to activate the service. This sequence must be unique and follow the syntax for digit maps (see [“Chapter 21 - Digit Maps” on page 293](#)). Dialing this digit map does not have any effect unless the service's status is “enabled”.
The activating sequence is set for all the lines of the Mediatrix 4100. You cannot have a different sequence for each line.
4. Define the digits that users must dial to stop the service in the *subscriberServicesCallForwardOnBusyDisableDigitMap* variable (under the *subscriberServicesActivationDigitmaps* group).
Define this variable only if you want to let the user stop the service with his or her handset. If you rather want to have the control over the activation/deactivation of the service, see Step 2.
For instance, you could decide to put “*73” as the sequence to deactivate the service. This sequence must be unique and follow the syntax for digit maps (see [“Chapter 21 - Digit Maps” on page 293](#)). Dialing this digit map does not have any effect unless the service's status is “enabled”.
The deactivating sequence is set for all the lines of the Mediatrix 4100. You cannot have a different sequence for each line.
5. Define the address to which forward incoming calls in the *subscriberServicesCallForwardOnBusyForwardingAddress* variable.
Accepted formats are:
 - telephone numbers (5551111)
 - SIP URLs such as “scheme:user@host”. For instance, “sip:user@foo.com”.
 This string is used literally, so cosmetic symbols (such as the dash in “555-xxxx”) should not be present.
Because this variable is located in a table, you can have a different string for each line.
6. Enable the Call Forward On Busy by setting the *subscriberServicesCallForwardOnBusyEnable* variable to **enable** (under the *subscriberServicesIfEnablingTable* group).
If you set the variable to **disable**, this will not disable the call forward, but will prevent the user from activating or deactivating the service. The user will not be able to use the digits used to start and stop the service.
Because this variable is located in a table, you can enable/disable the service on a per-line basis.

Using Call Forward on Busy

The following is the procedure to use this service on the user's telephone.

► To forward calls:

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence the system administrator has implemented to activate the call forward on busy service.
This sequence could be something like *72.
4. Wait for the transfer tone (three "beeps") followed by the dial tone.
5. Dial the number to which you want to forward your calls. Dial any access code if required.
6. Wait for three "beeps" followed by a silent pause.
The call forward is established.
7. Hang up your telephone.
The calls are checked against the digit maps set up by the system administrator.

► To cancel the call forward:

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence the system administrator has implemented to deactivate the call forward on busy service.
This sequence could be something like *73.
4. Wait for three "beeps" followed by a silent pause.
The call forward is cancelled.
5. Hang up your telephone.

On No Answer

You can forward the incoming calls of your users to a pre-determined target if they do not answer their telephone before a specific amount of time. The user does not have any feedback that a call was forwarded.

Setting up Call Forward On No Answer

You must configure and enable this service before your users can use it.

► **To set the Call Forward On No Answer feature:**

1. In the *subscriberServicesMIB*, locate the *subscriberServicesIfCallForwardActivationTable* group.
2. Set the status of the service in the *subscriberServicesCallForwardOnNoAnswerActivation* variable to **inactive** or **active**.
This variable starts the service (active) or stops the service (inactive).
If you want to let the user start or stop the service with his or her handset, you must enter a sequence of digits in steps 3 and 4. The *subscriberServicesCallForwardOnNoAnswerActivation* variable is automatically updated to reflect the activation status according to the user's setting.
3. Define the digits that users must dial to start the service in the *subscriberServicesCallForwardOnNoAnswerEnableDigitMap* variable (under the *subscriberServicesActivationDigitmaps* group).
Define this variable only if you want to let the user start the service with his or her handset. If you rather want to have the control over the activation/deactivation of the service, see Step 2.
For instance, you could decide to put “*74” as the sequence to activate the service. This sequence must be unique and follow the syntax for digit maps (see [“Chapter 21 - Digit Maps” on page 293](#)). Dialing this digit map does not have any effect unless the service's status is “enabled”.
The activating sequence is set for all the lines of the Mediatrix 4100. You cannot have a different sequence for each line.
4. Define the digits that users must dial to stop the service in the *subscriberServicesCallForwardOnNoAnswerDisableDigitMap* variable (under the *subscriberServicesActivationDigitmaps* group).
Define this variable only if you want to let the user stop the service with his or her handset. If you rather want to have the control over the activation/deactivation of the service, see Step 2.
For instance, you could decide to put “*75” as the sequence to deactivate the service. This sequence must be unique and follow the syntax for digit maps (see [“Chapter 21 - Digit Maps” on page 293](#)). Dialing this digit map does not have any effect unless the service's status is “enabled”.
The deactivating sequence is set for all the lines of the Mediatrix 4100. You cannot have different sequences for each line.
5. Define the address to which forward incoming calls in the *subscriberServicesCallForwardOnNoAnswerForwardingAddress* variable.
Accepted formats are:
 - telephone numbers (5551111)
 - SIP URLs such as “scheme:user@host”. For instance, “sip:user@foo.com”.
 This string is used literally, so cosmetic symbols (such as the dash in “555-xxxx”) should not be present.
Because this variable is located in a table, you can have a different string for each line.
6. Define the time, in milliseconds, the telephone keeps ringing before the call forwarding activates in the *subscriberServicesCallForwardOnNoAnswerTimeout* variable.
The default value is 5000.
7. Enable the Call Forward On No Answer by setting the *subscriberServicesCallForwardOnNoAnswerEnable* variable to **enable** (under the *subscriberServicesIfEnablingTable* group).

If you set the variable to **disable**, this will not disable the call forward, but will prevent the user from activating or deactivating the service. The user will not be able to use the digits used to start and stop the service.

Because this variable is located in a table, you can enable/disable the service on a per-line basis.

Using Call Forward on No Answer

The following is the procedure to use this service on the user's telephone.

► To forward calls:

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence the system administrator has implemented to activate the call forward on no answer service.
This sequence could be something like *74.
4. Wait for the transfer tone (three "beeps") followed by the dial tone.
5. Dial the number to which you want to forward your calls. Dial any access code if required.
6. Wait for three "beeps" followed by a silent pause.
The call forward is established.
7. Hang up your telephone.
The calls are checked against the digit maps set up by the system administrator.

► To cancel the call forward:

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence the system administrator has implemented to deactivate the call forward on no answer service.
This sequence could be something like *75.
4. Wait for three "beeps" followed by a silent pause.
The call forward is cancelled.
5. Hang up your telephone.

Call Waiting

The call waiting tone indicates to an already active call that a new call is waiting on the second line.

Your users can activate/deactivate the call waiting tone for their current call. This is especially useful when transmitting faxes. The user that is about to send a fax can thus deactivate the call waiting tone to ensure that the fax transmission will not be disrupted by an unwanted second call. When the fax transmission is completed and the line is on-hook, the call waiting tone is automatically reactivated.

The user can also permanently activate/deactivate the call waiting service.

Furthermore, the Mediatrix 4100 supports receiving some Call Waiting control commands via the SIP INFO method. See [“Controlling the Call Waiting Tone via SIP INFO” on page 283](#) for more details.

Setting up Call Waiting

You must configure and enable this service before your users can use it.

► **To set the Call Waiting service:**

1. In the *subscriberServicesMIB*, locate the *subscriberServicesIfEnablingTable* group.
2. Enable the Call Waiting feature by setting the *subscriberServicesCallWaitingEnable* variable to **enable**.
 This permanently activates the call waiting tone. When receiving new calls during an already active call, a special tone is heard to indicate that a call is waiting on the second line. The user can then answer that call by using the “flash” button. The user can switch between the two active calls by using the “flash” button.
 The call hold service must be enabled for this service to work. See [“Call Hold” on page 306](#).
 If the user is exclusively using faxes, put the variable to **disable** to permanently disable the call waiting tone.
 Because this variable is located in a table, you can enable/disable the service on a per-line basis. The user may cancel this service on a per-call basis when dialing a DTMF sequence matching the digit map stored in *subscriberServicesCallWaitingCancelDigitMap* (see Step 3). The user may also disable or enable this service permanently with the *subscriberServicesCallWaitingPermanentDigitMapEnable* and *subscriberServicesCallWaitingPermanentDigitMapDisable* digit maps (See Step 4).
 You can find the current status of the service in the *subscriberServicesCallWaitingStatus* read-only variable (under the *subscriberServicesIfStatusTable*).
3. Define the digits that users must dial to disable the Call Waiting tone in the *subscriberServicesCallWaitingCancelDigitMap* variable.
 This allows a user who has call waiting enabled to disable that service on the next call only. If, for any reason, the user wishes to undo the cancel, simply unhook and re-hook the telephone to reset the service.
 For instance, you could decide to put “*76” as the sequence to disable the call waiting tone. This sequence must be unique and follow the syntax for digit maps (see [“Chapter 21 - Digit Maps” on page 293](#)). Dialing this digit map does not have any effect unless the service's status is “enabled”. The deactivating sequence is set for all the lines of the Mediatrix 4100. You cannot have a different sequence for each line.
4. Define the digits that users must dial to enable the call waiting service permanently in the *subscriberServicesCallWaitingPermanentDigitMapEnable* variable.
 This activation is permanent until the user deactivates the service as in Step 5.
 For instance, you could decide to put “*84” as the sequence to enable the service. This sequence must be unique and follow the syntax for digit maps (see [“Chapter 21 - Digit Maps” on page 293](#)). The sequence is set for all the lines of the Mediatrix 4100. You cannot have a different sequence for each line.

When dialing this digit map, this sets the *subscriberServicesCallWaitingEnable* variable for the line the user is currently using to **enable**.

5. Define the digits that users must dial to disable the call waiting service permanently in the *subscriberServicesCallWaitingPermanentDigitMapDisable* variable.

This deactivation is permanent until the user enables the service as in Step 4.

For instance, you could decide to put “*85” as the sequence to disable the service. This sequence must be unique and follow the syntax for digit maps (see [“Chapter 21 - Digit Maps” on page 293](#)).

The sequence is set for all the lines of the Mediatrix 4100. You cannot have a different sequence for each line.

When dialing this digit map, this sets the *subscriberServicesCallWaitingEnable* variable for the line the user is currently using to **disable**.

Using Call Waiting

The call waiting feature alerts the user if he or she is already on the telephone and a second call happens. A “beep” (the call waiting tone) is heard and repeated every ten seconds to indicate there is a second incoming call.

► To put the current call on hold:

1. Perform a Flash-Hook by pressing the “Flash” button on your analog telephone.
This puts the call on hold and the second line is automatically connected to your line.
2. Answer the call on the second line.

► To switch from one line to the other:

1. Perform a Flash-Hook each time you want to switch between lines.

► To terminate the first call before answering the second call:

1. Hang up the telephone.
2. Wait for the telephone to ring.
3. Answer the telephone.
The second call is on the line.

► To terminate the active call and recover the call on hold with the flash hook and digit method:

1. Perform a Flash-Hook by pressing the “Flash” button on your analog telephone.
2. Dial the digit 1.

► To hold the active call and recover the call on hold with the flash hook and digit method:

1. Perform a Flash-Hook by pressing the “Flash” button on your analog telephone.
2. Dial the digit 2.

Removing the Call Waiting Tone

You can temporarily activate/deactivate the call waiting tone indicating a call is waiting. This is especially useful when transmitting faxes. If you are about to send a fax, you can thus deactivate the call waiting tone to ensure that the fax transmission is not disrupted by an unwanted second call. When the fax transmission is completed and the line is on-hook, the call waiting tone is automatically reactivated.

► To deactivate the call waiting tone:

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence the system administrator has implemented to deactivate the call waiting tone.

This sequence could be something like *70.

4. Wait for the transfer tone (three “beeps”) followed by the dial tone.
The call waiting tone is disabled.

► **To re-enable the call waiting tone:**

1. Take the receiver off-hook.
2. Replace the receiver on-hook.
The call waiting tone is re-enabled.

Permanently Removing the Call Waiting Tone

You can permanently activate/deactivate the call waiting service.

► **To activate the call waiting service:**

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence the system administrator has implemented to activate the call waiting tone service.
This sequence could be something like *84.
4. Wait for the transfer tone (three “beeps”) followed by the dial tone.
5. Hang up your telephone.
The call waiting tone is enabled.

► **To cancel the call waiting service:**

1. Take the receiver off-hook.
2. Wait for the dial tone.
3. Dial the sequence the system administrator has implemented to deactivate the call waiting tone service.
This sequence could be something like *85.
4. Wait for the transfer tone (three “beeps”) followed by the dial tone.
The call waiting is cancelled.
5. Hang up your telephone.

Call Transfer

The Call Transfer service offers various ways to transfer calls:

- ▶ Blind Transfer
- ▶ Attended Transfer

The SIP protocol also offers to set transfer-related parameters. See [“Call Transfer Capacity” on page 271](#) and [“Referred-By Field” on page 274](#) for more details.

Blind Transfer

The blind call transfer service is sometimes called Transfer without Consultation or Unattended Transfer. It allows a user to transfer a call on hold to a still ringing (unanswered) call. The individual at the other extension or telephone number does not need to answer to complete the transfer.

The call hold and second call services must be enabled for this service to work. See [“Call Hold” on page 306](#) and [“Second Call” on page 307](#).

Enabling Blind Call Transfer

You must enable this service before your users can use it.

▶ **To enable the blind transfer service:**

1. In the *subscriberServicesMIB*, locate the *subscriberServicesIfEnablingTable* group.
2. Set the *subscriberServicesBlindTransferEnable* variable to **enable**.
Because this variable is located in a table, you can enable/disable the service on a per-line basis. You can find the current status of the service in the *subscriberServicesBlindTransferStatus* read-only variable (under the *subscriberServicesIfStatusTable*).

Using Blind Call Transfer

The following is the procedure to use this service on the user's telephone.

▶ **To transfer a current call blind:**

1. Perform a Flash-Hook by pressing the “Flash” button on your analog telephone. This puts the call on hold.
2. Wait for the transfer tone (three “beeps”).
3. Dial the number to which you want to transfer the call.
4. Wait for the ringback tone, then hang up your telephone.
The call is transferred. You can also wait for the third party to answer if you want. In this case, the call transfer becomes attended.
If you want to get back to the first call (the call on hold), you must perform two Flash-Hooks. You are back with the first call and the third party is released.

▶ **To transfer a call on hold with the flash hook and digit method:**

1. Perform a Flash-Hook by pressing the “Flash” button on your analog telephone.
2. Dial the digit 4.

Attended Transfer

The attended call transfer service is sometimes called Transfer with Consultation. It allows a user to transfer a call on hold to an active call. The individual at the other extension or telephone number must answer to complete the transfer.

The call hold and second call services must be enabled for this service to work. See [“Call Hold” on page 306](#) and [“Second Call” on page 307](#).

Enabling Attended Call Transfer

You must enable this service before your users can use it.

► **To enable the attended transfer service:**

1. In the *subscriberServicesMIB*, locate the *subscriberServicesIfEnablingTable* group.
2. Set the *subscriberServicesAttendedTransferEnable* variable to **enable**.
Because this variable is located in a table, you can enable/disable the service on a per-line basis.
You can find the current status of the service in the *subscriberServicesAttendedTransferStatus* read-only variable (under the *subscriberServicesIfStatusTable*).

Using Attended Call Transfer

The following is the procedure to use this service on the user's telephone.

► **To transfer a current call attended:**

1. Perform a Flash-Hook by pressing the “Flash” button on your analog telephone.
This puts the call on hold.
2. Wait for the transfer tone (three “beeps”).
3. Dial the number to which you want to transfer the call.
The third party answers.
4. Hang up your telephone.
The call is transferred.
5. If you want to get back to the first call (the call on hold), you must perform two Flash-Hooks.
You are back with the first call and the third party is released.



Note: If the number to which you want to transfer the call is busy or does not answer, quickly perform a Flash-Hook. The busy tone or ring tone is cancelled and you are back with the first call.

► **To transfer a call on hold with the flash hook and digit method:**

1. Perform a Flash-Hook by pressing the “Flash” button on your analog telephone.
2. Dial the digit 4.

Conference Call

Standards Supported

RFC 4579 – Session Initiation Protocol (SIP) Call Control - Conferencing for User Agents

The Conference Call service allows a user to link two or more calls together to form a single conversation, called a conference.

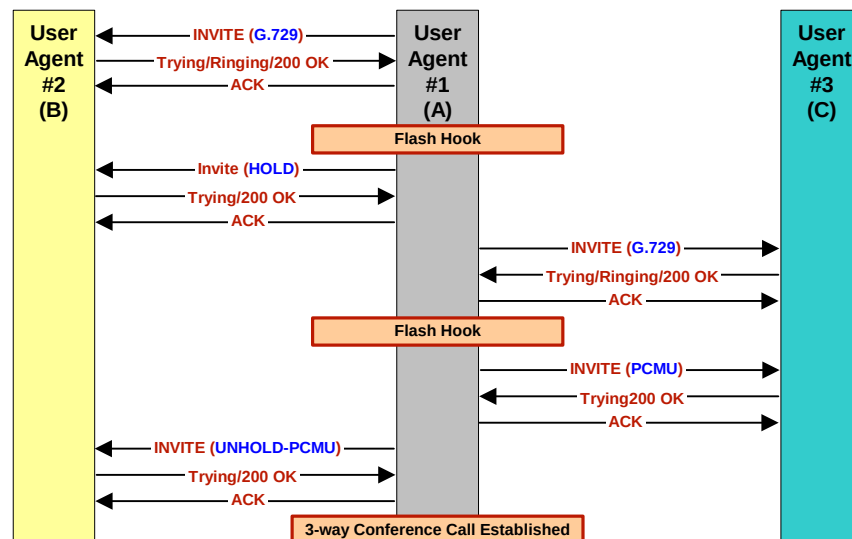
A participant of the conference can put the conference on hold and attempt other calls. This participant may then rejoin the conference at a later time by unholding it. The participant who initiated the conference cannot put it on hold.

You must enable the call hold and second call services for this service to work. See [“Call Hold” on page 306](#) and [“Second Call” on page 307](#).

Furthermore, you must also enable the Attended Transfer service for the two other participants to stay connected once the participant who initiated the conference has hung up. See [“Attended Transfer” on page 318](#).

The following is a conference call flow example:

Figure 90: Conference Call Flow



Requirements

For the conference call to occur successfully, all parties must meet the following requirements:

- ▶ Support at least one of the PCM codecs (G.711 μ -law and G.711 A-law) enabled on the line that is having the conference. See [“Enabling Individual Codecs” on page 233](#) for more details.
- ▶ Ability to dynamically change codec during a call.
- ▶ The packetization period (ptime) should be the same for all the participants of the conference. If this is not the case, then part of the conversation may be lost, resulting in a choppy voice. For better results, Media5 recommends to set the packetization period of all participants of a 3-way conference to 30 milliseconds. See [“Packetization Time” on page 234](#) for more information on how to set the packetization period of the Mediatix 4100.

Enabling the Conference Call Feature

You must enable this service before your users can use it.

► **To enable the conference call service:**

1. In the *subscriberServicesMIB*, locate the *subscriberServicesIfEnablingTable* group.
2. Set the *subscriberServicesConferenceEnable* variable to **enable**.
Because this variable is located in a table, you can enable/disable the service on a per-line basis. You can find the current status of the service in the *subscriberServicesConferenceStatus* read-only variable (under the *subscriberServicesIfStatusTable*).
3. In the *sipMIB*, locate the *sipConferenceConfigTable* group.
4. Specify how to manage a SIP conference in the *sipConferenceType* variable.
This configuration only applies to a conference initiated by one of the unit's endpoint.

Table 210: Conference Type Parameters

Parameter	Description
local	The conference is managed locally by the unit. The maximum number of participants is 3. This is the default value.
conferenceServer	The conference is managed by a remote SIP conference server. When a Flash-Hook occurs, the server mechanism is determined by the variable <i>sipInteropConferenceServerMecanism</i> (see Step 6). When using this conference type, both the initiator and a participant of the conference can add new participants to the conference.

This variable only has an effect when *subscriberServicesConferenceEnable* is set to enable.

Because this variable is located in a table, you can set it on a per-line basis.

5. If you have selected **conferenceServer** in the previous step, set the *sipConferenceServerURI* variable with the address of a conference server.
The format must be a SIP URI such as "scheme:user@host". For instance, "sip:user@foo.com".
Because this variable is located in a table, you can have a different string for each line.
6. If you have selected **conferenceServer** in Step 4, set the *sipInteropConferenceServerMechanism* variable (in the *sipInteropMIB*) with the mechanism used to establish a conference and how the participants are connected to the conference server.

Table 211: Conference Server Mechanism

Parameter	Description
rfc4579WithoutErrorRecovery	The connection with the conference server is made as defined in RFC 4579. The participants are connected to the conference server by sending the REFERs simultaneously. The connection with the participant is terminated if the participant fails to complete the REFER.
rfc4579WithErrorRecovery	The connection with the conference server is made as defined in RFC 4579. The participants are connected to the conference server by sending the REFERs sequentially. The REFER is not sent to the second participant and the call stays in the same state if the first participant fails to complete the REFER. This is the default value.

Managing a Conference Call

If you are on the telephone with one person and want to conference with a third one, you can do so. In the following examples, let's assume that:

- ▶ "A" is the conference initiator.
- ▶ "B" is the person called on the first line.
- ▶ "C" is the person called on the second line.
- ▶ "D" is a fourth person that "A" wants to add to the conference in **conferenceServer** conference type.

▶ **To initiate a conference ("A" and "B" already connected):**

1. "A" performs a Flash-Hook.
This puts "B" on hold and the second line is automatically connected. "A" hears a dial tone.
2. "A" dials "C's" number.
"A" and "C" are now connected.
3. "A" performs another Flash-Hook.
The call on hold ("B") is reactivated. "A" is now conferencing with "B" and "C".

▶ **To initiate a conference with the flash hook and digit method ("A" and "B" already connected):**

1. "A" performs a flash hook.
This puts "B" on hold and the second line is automatically connected. "A" hears a dial tone.
2. "A" dials "C's" number.
The context is now one call active and one call on hold.
3. "A" performs a flash hook, and then dials the digit **3** to start the conference call.
The call on hold ("B") is reactivated. "A" is now conferencing with "B" and "C".



Note: Performing a flash hook and dialing the digit **2** will stop conference but keep one active call and one call on hold.

▶ **"A" wants to transfer "B" to "C" during the conference:**

This is available only in the **local** conference type.

1. "A" hangs up.
The conference is terminated. "B" and "C" are now connected.

▶ **"A" wants to terminate the call with "C" and get back to the call with "B" during the conference:**

This is available only in the **local** conference type.

1. "A" performs a Flash-Hook.
The conference is terminated and the call with "C" is disconnected. "A" and "B" are still connected and can go on with their conversation.

▶ **"B" (or "C") hangs up during the conference:**

This is available only in the **local** conference type.

1. "B" (or "C") hangs up during the conference.
The conference is terminated, but the call between "A" and "C" (or "B") is not affected and they are still connected.

▶ **"A" wants to add a fourth member to the conference:**

This is available only in the **conferenceServer** conference type.

1. "A" performs a Flash-Hook.
This puts "B" and "C" on hold and the second line is automatically connected. "A" hears a dial tone.

2. "A" dials "D's" number.
"A" and "D" are now connected.
3. "A" performs another Flash-Hook.
The call on hold ("B" and "C") is reactivated. "A" is now conferencing with "B", "C", and "D".

The telephony attributes are used to configure the characteristics of the telephony system being implemented.

Automatic Call

The automatic call feature allows you to define a telephone number that is automatically dialed when taking the handset off hook.



In the *Unit Manager Network Administration Manual*, refer to chapter *Telephony Attributes Parameters*, section *Telephony Attributes Configuration Window*.

When this service is enabled, the second line service is disabled but the call waiting feature is still functional. The user can still accept incoming calls.

You can also set these parameters via the web interface, as described in [“Automatic Call” on page 103](#).

► **To set the automatic call feature:**

1. In the *telephonyAttributesMIB*, locate the *telephonyAttributesIfFeaturesTable* group.

This group contains all of the variables required to set the automatic call feature.

2. Define the number to dial when the handset is taken off hook in the *telephonyAttributesAutomaticCallTargetAddress* variable.

Accepted formats are:

- telephone numbers (5551111)
- SIP URLs such as "scheme:user@host". For instance, "sip:user@foo.com".

This string is used literally, so cosmetic symbols (such as the dash in "555-xxxx") should not be present.

Because this variable is located in a table, you can define a different number for each line of the Mediatrix 4100.

3. Enable the automatic call feature by setting the *telephonyAttributesAutomaticCallEnable* variable to **enable**.

Because this variable is located in a table, you can enable/disable the feature on a per-line basis.

Call Direction Restriction

You can define in which direction calls are allowed.



In the *Unit Manager Network Administration Manual*, refer to chapter *Telephony Attributes Parameters*, section *Telephony Attributes Configuration Window*.

► **To set call direction restriction:**

1. In the *telephonyAttributesMIB*, locate the *telephonyAttributesIfFeaturesTable* group.
2. Define the restriction on the direction of traffic in the *telephonyAttributesCallDirectionRestriction* variable.

Table 212: Call Direction Restrictions

Restriction	Description
noRestriction	Allows incoming and outgoing calls.
scnTolpOnly	The Mediatrix 4100 allows to make calls but cannot receive calls.
ipToScnOnly	The Mediatrix 4100 allows to receive calls but does not allow to make calls.

Because this variable is located in a table, you can define a different call direction for each line of the Mediatrix 4100.

Hook Flash Processing

Standards Supported

- RFC 2976: The SIP INFO Method

Hook flash processing allows hook flash signals to be transported over the IP network allowing to use advanced telephony services. You can define how to process hook-flash detection. Users normally press the “flash” button of the telephone during a call in progress to put this call on hold, transfer it, or even initiate a conference call.



Note: The hook flash processing attribute is not negotiated in SDP.

► **To define how to process hook flash:**

1. In the *telephonyAttributesMIB*, set the *telephonyAttributesHookFlashProcessing* variable. This allows the enabled subscriber services to be handled by the unit or to be delegated to a remote party.

Table 213: Hook Flash Settings

Setting	Definition
processLocally	The hook-flash is processed locally. The actual behaviour of the “flash” button depends on which subscriber services are enabled for this line. See “Chapter 23 - Subscriber Services” on page 305 for more details.

Table 213: Hook Flash Settings (Continued)

Setting	Definition
transmitUsingSignalingProtocol	The hook-flash is processed by a remote party. The hook-flash event is carried by a signaling protocol message. The actual behaviour of the “flash” button depends on the remote party. The hook-flash event is relayed as a SIP INFO message as described in RFC 2976. Note: This feature and the DTMF relay feature via signalling protocol are totally independent. Activating one of these features has no effect on the other. See “DTMF Transport Type” on page 237 for more details. Note: This setting disables all subscriber services that use the “flash” button, such as the Call Hold service.
outOfBandUsingRtp	The hook-flash is processed by a remote party. The hook-flash event is relayed as telephone-event 16 via an RFC 2833 RTP packet. The actual behavior of the 'flash' button depends on the remote party.

IP Address Call Service

The IP address call service allows a user to dial an IP address without the help of a SIP server. Using this method bypasses any server configuration of your unit.

The user can dial an IP address and enter an optional telephone number. Note that the optional telephone number is matched by using the same digit maps as a normal call.

Enabling IP Address Calls

► To enable the IP address call service:

1. In the *telephonyAttributesMIB*, locate the *telephonyAttributesIpAddressCallCustomization* group.
2. Enable the IP address call service by setting the *telephonyAttributesIpAddressCallEnable* variable to **enable**.

Dialing an IP Address

► To make an IP address call:

1. Dial “**” (IP address prefix).
2. Dial the numerical digits of the IP address and use the “*” for the “.” of the IP address.
3. Dial “#” to terminate the IP address.
4. Dial the telephone number of the specific line you want to reach.

For example, let's say you want to reach the telephone connected to Line 2 of the Mediatrix 4100 with the IP address 192.168.0.23. The phone number assigned to Line 2 of this Mediatrix 4100 is 1234. You must then dial the following digits:

****192*168*0*23#1234**

In this case, the Mediatrix 4100 sends an INVITE *1234@192.168.0.23*.

PIN Dialing

Standards Supported

draft-choudhuri-sip-info-digit-00.txt

The PIN Dialing feature allows you to configure a PIN (Personal Identification Number) that would be dialed “n” milliseconds after an outgoing call was established.

This feature could be used in the case where a user makes an automatic call to an IVR system, and after a pre-defined delay, the Mediatrix 4100 sends the DTMF tones (PIN) to indicate where the call is coming from.

The PIN is transmitted by using the DTMF out-of-band by signalling protocol transport type. Both parties involved must thus support the *draft-choudhuri-sip-info-digit-00.txt* draft. The PIN must be negotiated in the call. See [“DTMF Transport Type” on page 237](#) for more details on the DTMF out-of-band by signalling protocol transport type.

► **To configure the PIN dialing feature:**

1. In the *pinDialingMIB*, define the PIN to dial in the *pinDialingPin* variable.
The PIN contains the DTMFs to be dialed. The supported digits are “0123456789*#abcdABCD”.
Pause characters “,”, “;”, and “p” are also supported and represent 1 second.



Note: The *draft-choudhuri-sip-info-digit-00.txt* draft does not support the pause characters “,”, “;”, and “p”. This is a proprietary support.

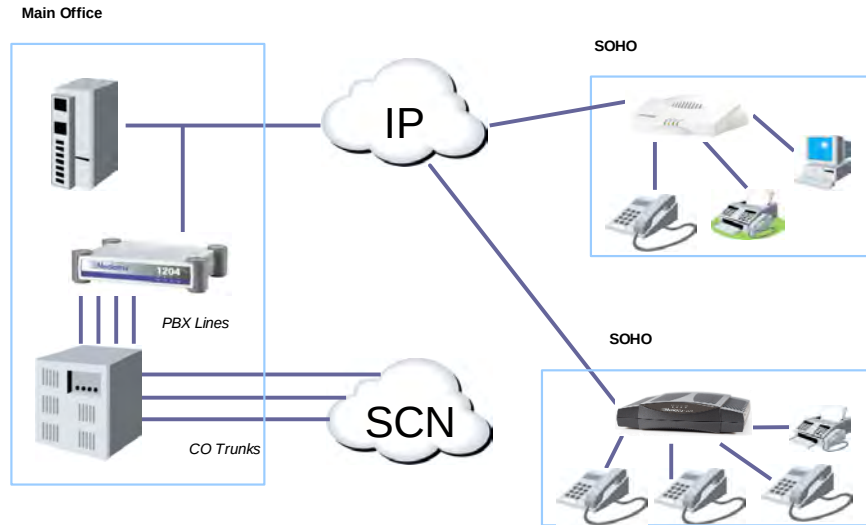
2. Set the delay prior to sending the PIN in the *pinDialingDelay* variable.
This value is expressed in milliseconds (ms). The default value is 1000 ms.
3. Enable the PIN dialing feature by setting the *pinDialingEnable* variable to **enable**.

Remote Line Extension

The Remote Line Extension feature makes it possible to connect remote small offices with similar capabilities as if they were located in the main or head office. Essential features such as integrated voicemail, unified messaging, line extension dialing plans and many others can be offered to remote sites.

Mediatix FXS access devices and Mediatix 1204 FXO gateways extend PBX extensions to remote workers located at SOHOs (Small Offices/Home Offices), using VoIP. PBX extensions are connected via Mediatix 1204 FXO ports to the IP network, instead of being connected to individual stations. At the SOHO locations, Mediatix FXS units connect analog phones to the same IP network.

Figure 91: Remote Line Extensions



► **To configure Remote Line Extensions:**

1. Set the Mediatix 1204 lines to perform automatic calls to a pre-defined number as defined in ["Automatic Call" on page 323](#).
This allows you to define a telephone number that is automatically dialed when seizing an FXO line. You can thus redirect SCN calls to a specific IP number such as a telephone connected to a FXS port of a Mediatix FXS device.
2. If applicable, you can instruct the Mediatix 1204 to wait until the called party answers the phone before it picks up the SCN line. You can do so in the *fxoMIB* by setting the *fxoWaitForCalleeToAnswerEnable* variable to **enable**.
3. If applicable, define how the Mediatix 1204 behaves when it receives an IP call in the *analogScnGwDialEnable* variable.

Table 214: IP Call Reception Behaviour

Parameter	Description
disable	The Mediatix 4100 picks up the SCN line and opens the audio channel with the peer. This way, the user of a dedicated FXS/FXO combinaison will feel "closer" to the SCN: as soon as the user takes the receiver off-hook, he/she interacts with the SCN because the actions are not relayed via IP to the FXO unit.
enable	The Mediatix 4100 picks up the SCN line and dials the telephone number associated with the destination of the call before opening the audio channel with the peer. This is the default value.

4. Define the Hook Flash Processing as per [“Hook Flash Processing” on page 324](#). Setting the *telephonyAttributesHookFlashProcessing* variable to **outOfBandUsingRtp** means the following:

Table 215: Hook Flash Processing

Mediatrix 1204	Mediatrix FXS Device
The hook-flash event received via an RFC 2833 RTP packet is executed.	The hook-flash event is relayed as telephone-event 16 via an RFC 2833 RTP packet.

5. Set the behaviour for the support of RFC 2833 in the *voicelfDtmfEnforceDefaultEvents* variable for both units.

Table 216: DTMF Enforce Default Events

Parameter	Description
enable	Conformance is enforced and support for RFC 2833 implies the support of basic telephony-events. When setting the variable <i>voicelfDtmfTransport</i> to outOfBandUsingRtp (“DTMF Transport Type” on page 237), or the variable <i>telephonyAttributesHookFlashProcessing</i> to outOfBandUsingRtp (“Hook Flash Processing” on page 324), the unit will advertise the support for events 0-15; it will assume support for events 0-15 when support for RFC 2833 is received in an announcement.
disable	This creates a deliberate deviance to RFC 2833 as support of basic events is not automatic. The variables <i>voicelfDtmfTransport</i> and <i>telephonyAttributesHookFlashProcessing</i> then act independently to specify which events will be relayed via RFC 2833. If Hook Flash relay is enabled by itself, support of event 16 alone will be advertised; if both Hook Flash and DTMF relay are activated, events 0-16 are supported.

6. If applicable, configure port mapping as per [“Source Line Selection” on page 225](#). You could map FXO ports to IP Phones or analog phones connected to FXS ports. This creates transparent and user-friendly call scenarios, as IP endpoints can act as remote PBX extensions. Thus the reach of a PBX can be extended beyond the physical location of the PBX. This is especially an attractive option for SOHO users.

Delayed Hot Line

The delayed hot line feature is used to make an automatic call to a specified address on the two following conditions:

- ▶ When the user picks up the phone but does not dial any digit.
- ▶ When the user starts dialing but does not complete a valid number before the timeout set in the *digitMapTimeoutCompletion* variable expires. This is the delayed hotline extension feature.

This feature thus places an automatic call whenever the *digitMapTimeoutFirstDigit* timeout expires. It could be used as an alternative to the emergency number (for instance, the 911 number in North America).

▶ To configure the basic delayed hot line feature:

1. Enable the delayed hot line feature by setting the *telephonyAttributesDelayedHotLineEnable* variable to **enable**.

When the feature is disabled, a user picking up the phone but not pressing any telephone keys hears the Receiver Off-Hook tone after the amount of time specified in the *digitMapTimeoutFirstDigit* variable.

► **To configure the delayed hot line extension feature:**

1. In the *telephonyAttributesMIB*, set the destination (address or telephone number) that is automatically called in the *telephonyAttributesDelayedHotLineTargetAddress* variable.
2. Enable or disable the Delayed Hot Line extension feature in the *telephonyAttributesDelayedHotLineExtension* variable.
With this extension, the automatic call mentioned in the basic feature is placed upon expiration of the *digitMapTimeoutCompletion* timer.

Table 217: Delayed Hot Line Extension Parameters

Parameter	Description
enable	The destination specified in Step 2 is called upon expiration of the timeout.
disable	A user beginning to dial a telephone number but failing to complete the operation before expiration of the <i>digitMapTimeoutCompletion</i> timeout hears the Receiver Off-Hook tone.

Call Rejection

An incoming call can be rejected if it rings without being answered for a certain amount of time. The default value, 0, indicates that an incoming ringing call is never rejected by the unit.

► **To configure the call rejection feature:**

1. In the *telephonyAttributesMIB*, set the number of seconds a port is allowed to ring before automatically rejecting the call in the *telephonyAttributesAutomaticRejection* variable.
A value of 0 means that there is no limit so the port rings until the caller hangs up.
2. In the *sipInteropMIB*, define the SIP code to use when rejecting a call once the timer specified in *telephonyAttributesAutomaticRejection* elapses in the *sipInteropAutomaticRejectionCode* variable.
Examples of possible SIP code:

400: Bad Request	480: Temporarily unavailable
401: Unauthorized	481: Call/Transaction Does not Exist
402: Payment required	482: Loop Detected
403: Forbidden	483: Too many hops
404: Not found	484: Address incomplete
405: Method not allowed	485: Ambiguous
406: Not acceptable	486: Busy here
407: Proxy authentication required	500: Server internal error
408: Request timeout	501: Not implemented
410: Gone	502: Bad gateway
413: Request Entity too long	503: Service unavailable
414: Request-URI too long	504: Server time-out
415: Unsupported media type	504: Version Not Supported
416: Unsupported URI Scheme	513: Message Too Large
420: Bad extension	600: Busy everywhere
421: Extension Required	603: Decline
423: Interval Too Brief	604: Does not exist anywhere

Message Waiting Indicator

This chapter explains how to set the Mediatrix 4100 to use the Message Waiting Indicator service.

You can also set these parameters via the web interface, as described in [“Message Waiting Indicator” on page 112](#).

What is Message Waiting Indicator (MWI)?

The Message Waiting Indicator (MWI) service alerts the user when new messages have been recorded on a voice mailbox.

When the user receives a call and does not answer, the notification mechanism detects this situation and starts the auto attendant. The caller can then leave a message.

After the message is recorded, the server sends a message to the Mediatrix 4100 listing how many new and old messages are available. The Mediatrix 4100 alerts the user of the new message in two different ways:

- ▶ The telephone's LED blinks (if present).
- ▶ A message waiting stutter dial tone replaces the normal dial tone when the user picks up the first line.



Note: The message waiting state does not affect the Second Line feature. When in an active call, performing a flash-hook to get access to the second line plays the usual dial tone.

Standard MWI Methods

The Mediatrix 4100 supports two MWI methods.

MWI Method #1

Standards Supported

- draft-ietf-sipping-mwi-01.txt (MWI draft)
- “Telecordia GR-1401-CORE (Issue 1, June 2000)” specification (visual message indication (LED blinking))
- “GR-506-CORE (Issue 1, with Revision 1, November 1996)” specification (message waiting indicator tone)

The Mediatrix 4100 sends SUBSCRIBE requests to the server for each line, unless there is no subscription address defined. The Mediatrix 4100 then waits for NOTIFY requests containing the relevant message waiting information.

▶ To configure the MWI:

1. In the *mwimib*, set the notification mechanism server address to which the Mediatrix 4100 subscribes in the *mwiconfigUserSubscriptionAddress* variable.

This mechanism notifies the Mediatrix 4100 when new messages are available. The address is a SIP URL such as “scheme:user@host”. For instance, “sip:user@foo.com”.

Because this variable is located in a table, you can define a different address for each line of the Mediatrix 4100.

2. Define the digits that users must dial to retrieve messages in the *mwifetchDigitMap* variable.
Dialing these digits initiates a call to the voice messaging system. For instance, you could decide to put “*50” as the sequence a user must dial to retrieve voice messages. This sequence must be unique and follow the syntax for digit maps (see [“Chapter 21 - Digit Maps” on page 293](#)). Dialing this digit map does not have any effect unless the service's status is “enabled”.
The activating sequence is set for all the lines of the Mediatrix 4100. You cannot have different sequences for each line.
3. Set the destination to call to retrieve messages in the *mwiconfigFetchAddress* variable.
The user typically initiates a call to the voice messaging system, and then uses an auto-attendant to get the messages. Available formats are:
 - telephone numbers (5551111)
 - SIP URLs such as “scheme:user@host”. For instance, “sip:user@foo.com”.
 This string is used literally, so cosmetic symbols (such as the dash in “555-xxxx”) should not be present.
Because this variable is located in a table, you can define a different destination for each line of the Mediatrix 4100.
4. Enable or disable the voltage message waiting indicator state in the *mwiconfigVoltageEnable* variable.
When new messages are awaiting attention on the messaging server, the Mediatrix 4100 uses the line voltage to power the LED on the telephone in addition to the default method.
This variable is effective only if the endpoint is configured to use the visual message waiting indicator (see Step 7).
5. Define the duration, in seconds, of dynamic subscription to a messaging service in the *mwipirationTime* variable.
6. In the *sipInteropMIB*, define how Message Waiting Indicator notifications must be validated in the *sipInteropMwiMessageSummaryValidation* variable.

Table 218: Message Waiting Indicator Notifications

Parameter	Description
enable	In order to have the Message Waiting Indicator activated, the SIP notification must meet all of the following requirements: a. 'Messages-Waiting' must be set to yes. b. The message waiting media type must contain 'voice-message'. c. The number of new messages must be greater than or equal to 1.
disable	In order to have the Message Waiting Indicator activated, the SIP notification must meet the following requirement: 'Messages-Waiting' must be set to yes.

Note that only Message Waiting notifications for an established subscription are affected. Message Waiting notifications without subscription always behave as described in *disable*.

7. Enable the MWI by setting the *mwiconfigActivation* variable to **enable**.
Because this variable is located in a table, you can enable/disable the service on a per-line basis.



Note: The MWI subscription refresh is not supported when the caller ID is DTMF-based, so modifying the variable *mwiconfigActivation* will have no effect.

► **To refresh the MWI subscription:**

1. In the *mwimib*, set the *mwisubscriptionCmdRefresh* variable.
Available values are:
 - noOp: No operation.

- refresh: Refresh message waiting subscriptions. All enabled endpoints unsubscribe themselves from the service and re-subscribe by using the current provisioning.

MWI Method #2

Standards Supported	draft-mahy-sip-message-waiting-02.txt (expired) with proprietary modifications
----------------------------	--

This method does not require any special settings or configuration.

MWI Notify Service

The Mediatrix 4100 offers the possibility to extend some key features to remote extensions located in Branch or Home Offices across the SCN.

This service is available only when using the IP Communication Server v3.1 product as a SIP Redirect server.

For instance, a designated analog voice mail system at a main site can provide voice mail for the home or branch office. The home office user is notified of the message waiting via a message waiting LED on the telephone or a special tone when picking up the telephone.

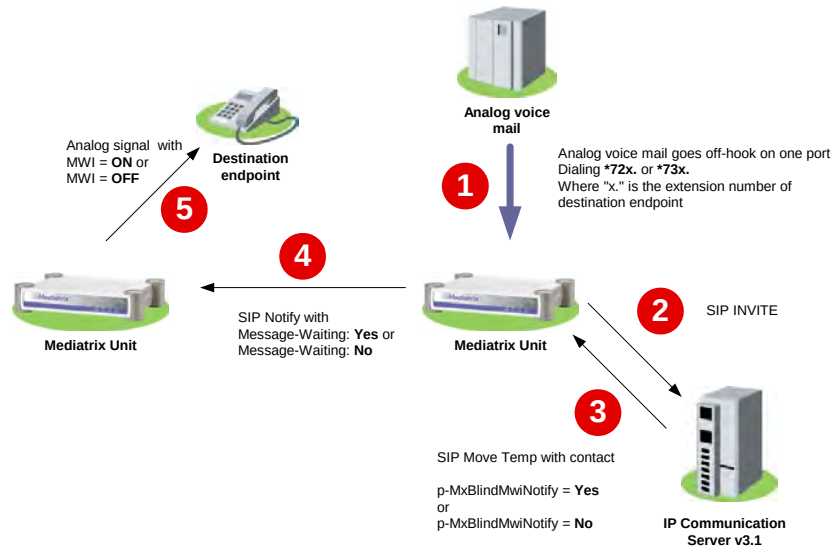
How does the Service Work?

The MWI Notify service is a proprietary feature. In this solution, the analog voice mail system is configured to seize a designated outgoing line and dial a pre-defined string such as “*72xxx” to notify the server it must give a message waiting indication to extension “xxx”. Once voice messages have been retrieved, the analog voice mail system seizes the designated outgoing line and dials a pre-defined string such as “*73xxx” to notify the server to turn off the message waiting indicator for extension “xxx”.

The service uses the Route Manager currently available in the IP Communication Server v3.1 to send a special command to the Mediatrix unit.

The following is the basic sequence of operations for the MWI Notify service:

1. The analog voice mail system dials the following digits:
*72101
where *72 is a prefix and 101 the user extension.
2. The Mediatrix unit sends a standard INVITE to the IP Communication Server v3.1 containing the complete dialed string (*72101).
 - a. The IP Communication Server looks for the registered user “*72101” in the Registrar database.
 - b. The IP Communication Server cannot find the user, so it asks the Route Manager to process the request.
 - c. Provided that the Route Manager is properly configured, it recognizes the “*72” prefix and associates it to the proper route conditions.
3. The IP Communication Server answers the request with a “Moved Temporarily”. It contains information about the target(s) in the *Contact* header plus a proprietary *p-MxBlindMWINotify=yes/no* field.
 - a. The Mediatrix unit retrieves the location from the IP Communication Server’s answer and the *p-MxBlindMWINotify* field.
 - b. The Mediatrix unit parses the answer from the IP Communication Server and recognizes *p-MxBlindMWINotify* as a special command.
4. The Mediatrix unit sends a NOTIFY to the location received from the IP Communication Server by using the proper yes or no value (*72 = yes, *73 = no) specified by the route condition.
5. The unit receiving the NOTIFY enables or disables the MWI service for the specified port/user.

Figure 92: Example of the MWI Notify Service

Configuring the IP Communication Server

In the Route Manager of the IP Communication Server, you must configure routes that would be triggered by a pre-defined prefix. The prefix could be any valid digits (DTMF). The example described above uses “*72” to enable the MWI and “*73” to disable the MWI.

For more information on how to configure the Route Manager, please refer to the *IP Communication Server Administration Manual* or the IP Communication Server contextual help.

Configuring the Mediatrix 4100

There is no special unit configuration required. The Mediatrix unit behaves as if in a standard call until it receives one of the following parameters in the *Contact* field:

- ▶ p-MxBlindMwiNotify=Yes
- or
- ▶ p-MxBlindMwiNotify=No

Upon receiving one of these parameters, the unit sends a NOTIFY to the destination endpoint instead of an INVITE. The sent NOTIFY is compliant with <draft-mahy-sip-message-waiting-02.txt>.

Management Server Configuration

The Management Server is a generic name for a module or software that is used to remotely set up Mediatrix 4100 units. For instance, the Management Server could be the Media5's Unit Manager Network product. See ["Unit Manager Network – Element Management System" on page xxii](#) for more details.



In the *Unit Manager Network Administration Manual*, refer to chapter *Administration Parameters*, section *Unit Manager Server*.

Using the Management Server

You have the choice of setting up Mediatrix 4100 units directly with a SNMP browser or with the Management Server. If you want to use the Management Server to setup the units, you shall tell these units how to reach the Management Server.

► To use the Management Server:

1. In the *msMIB*, locate the *msEnable* variable.
This variable enables the Management Server to remotely manage the Mediatrix 4100.
2. Set the *msEnable* variable to **enable**.
3. Set the Trap retransmission period (*msTrapRetransmissionPeriod* variable) to the desired value.
The available values range from 10 ms to 604 800 000 ms (1 week). The default value is 60 000 ms.
4. Set the Trap retransmission retry count (*msTrapRetransmissionRetryCount* variable) to the desired value.
When the retry count is elapsed, the Mediatrix 4100 stops the provisioning sequence. The default value is 10. If this variable is set to -1, then the provisioning sequence never stops. The trap is sent until the Management Server replies.

Configuration Source

The Mediatrix 4100 must know the IP address and port number of the Management Server. You can assign these information to the Mediatrix 4100 through a DHCP server or manually enter them yourself with the static variables.

DHCP Configuration

Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See ["Chapter 8 - IP Address and Network Configuration" on page 143](#) for more details.

► To use DHCP-assigned information:

1. In the *ipAddressConfig* folder, locate the *msSelectConfig Source* variable.
This variable defines whether the Mediatrix 4100 shall get its Management Server configuration through a DHCP server or not.
2. Set the *msSelectConfigSource* variable to **dhcp**.
You can query the Management Server's IP address and port number assigned by the DHCP server in the *msHost* and *msTrapPort* read-only variables (in the *ipAddressStatus* folder).

3. Set how you want to define the Management Server information in the DHCP server:

Table 219: Management Server DHCP Information

To use a...	Set...
vendor specific code	The <i>msDhcpSiteSpecificCode</i> variable to 0 . Set the management server IP address in the DHCP server inside the vendor specific sub-option 200 (hexadecimal 0xC8).
site specific code	The <i>msDhcpSiteSpecificCode</i> variable to any value between 128 and 254. Set the management server IP address in the DHCP server inside the site specific option you have chosen (it must match the value of the <i>msDhcpSiteSpecificCode</i> variable in the unit's configuration).

See ["Vendor and Site Specific DHCP Options" on page 156](#) for more details.

Static Configuration

Use the static configuration if you are not using a DHCP server or if you want to bypass it.

► To use static information:

1. In the *ipAddressConfig* folder, locate the *msSelectConfig Source* variable.
This variable defines whether the Mediatrix 4100 shall get its Management Server configuration through a DHCP server or not.
2. Set the *msSelectConfigSource* variable to **static**.
3. Set the following variables:

Table 220: Management Server Static Address

Variable	Description
msStaticHost	Static management server IP address or domain name. Default Value: 192.168.0.10
msStaticTrapPort	Static management server IP port number. Restart the unit to update this parameter. Default Value: 162 Note: Change the port used in the management server. Not doing so will prevent you from viewing the received traps from the unit. The management server could be a product such as the Unit Manager Network.

QoS (Quality of Service) features enable network managers to decide on packet priority queuing. The Mediatrix 4100 supports the Differentiated Services (DS) field and 802.1p taggings. There are three variables – one variable for signalling (SIP) and one variable for each of voice and T.38 media.

The Mediatrix 4100 supports the Real Time Control Protocol (RTCP), which is used to send packets to convey feedback on quality of data delivery.

The Mediatrix 4100 does not support RSVP (Resource Reservation Protocol).

Differentiated Services (DS) Field

Standards Supported

RFC 2475 – An Architecture for Differentiated Services

Differentiated Services (DiffServ, or DS) is a protocol for specifying and controlling network traffic by class so that certain types of traffic – for example, voice traffic, which requires a relatively uninterrupted flow of data, might get precedence over other kinds of traffic.

DiffServ replaces the first bits in the ToS byte with a differentiated services code point (DSCP). It uses the existing IPv4 Type of Service octet.

It is the network administrator's responsibility to provision the Mediatrix 4100 with standard and correct values.

You can also set these parameters via the web interface, as described in ["DiffServ Configuration" on page 119](#).

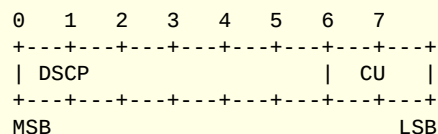
► To enable the DS field configuration:

1. In the *qosDiffServ* group of the *qosMIB*, locate the following variables:
 - *qosSignalingDiffServ*
 - *qosVoiceDiffServ*

What are Differentiated Services?

Differentiated Services avoids simple priority tagging and depends on more complex policy or rule statements to determine how to forward a given network packet. An analogy is made to travel services, in which a person can choose among different modes of travel – train, bus, airplane – degree of comfort, the number of stops on the route, standby status, the time of day or period of year for the trip, and so forth.

For a given set of packet travel rules, a packet is given one of 64 possible forwarding behaviors – known as per hop behaviors (PHBs). A six-bit field, known as the Differentiated Services Code Point (DSCP), in the Internet Protocol header specifies the per hop behavior for a given flow of packets. The DS field structure is presented below:



- *DSCP*: Differentiated Services CodePoint.
- *CU*: Currently Unused. The CU bits should always be set to 0.

For both signalling and media packets, the DSCP field is configurable independently. The entire DS field (TOS byte) is currently configurable.

- qosT38FaxDiffServ

These variables are 1 octet scalar ranging from 0 to 255. The DSCP default value should be 101110. This results in the DS field value of 10111000 (184).

This default value would result in a value of “101” precedence bits, low delay, high throughput, and normal reliability in the legacy IP networks (RFC 791, RFC 1812). Network managers of legacy IP networks could use the above-mentioned values to define filters on their routers to take advantage of priority queuing. The default value is based on the Expedited Forwarding PHB (RFC 2598) recommendation.



Note: RFC 3168 now defines the state in which to set the two least significant bits in the TOS byte. On the other hand, this RFC only applies to TCP transmissions and the bits are thus set to “0” in the Mediatix 4100. This has the following effects:

- The TOS values for UDP packets are the same as in the MIB.
- The TOS values for TCP packets are equal to the closest multiple of 4 value that is not greater than the value in the MIB.

2. Set the value you want to use.

You can find references on DS field under the IETF working group DiffServ. For more information, please refer to the following RFC documents:

- Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers (RFC 2474)
- An Architecture for Differentiated Services (RFC 2475)
- Assured Forwarding PHB Group (RFC 2597)
- An Expedited Forwarding PHB (RFC 2598)

IEEE 802.1q

The 802.1q standard recommends the use of the 802.1q VLAN tags for Ethernet frames traffic prioritization. VLAN tags are 4-byte headers in which three bits are reserved for priority indication. The values of the priority bits shall be provisioned.

The 802.1q standard comprises the 802.1p standard.

It is the network administrator's responsibility to provision the Mediatix 4100 with standard and correct values. You can also set these parameters via the web interface, as described in [“802.1q Configuration” on page 117](#).

► To enable the IEEE 802.1q user priority configuration:

1. In the *qosIEEE8021q* group of the *qosMIB*, locate the following variables:
 - qosSignalingIEEE8021qEnable
 - qosVoiceIEEE8021qEnable
 - qosT38FaxIEEE8021qEnable

2. Set the value of these variables to **enable**.
The corresponding user priority configuration is enabled.

3. In the *qosIEEE8021q* group of the *qosMIB*, locate the following variables:
 - qosSignalingIEEE8021qUserPriority
 - qosVoiceIEEE8021qUserPriority
 - qosT38FaxIEEE8021qUserPriority

These variables are 1 octet scalar ranging from 0 to 7. The 802.1q default priority value should be 6 for both signalling and media packets.

4. Set the value you want to use.

For more information, please refer to the *MIB Reference* manual.

Voice QoS vs RTCP Packets

You can define whether or not the configuration related to the voice QoS (*qosVoiceDiffServ* and *qosVoiceIeee8021qUserPriority* variables) is also applied to the RTCP packets generated by the device.

► **To define the voice QoS configuration behavior :**

1. In the *qosInterop* group of the *qosMIB*, locate the *qosInteropUseVoiceQoSForRtcpEnable* variable.
2. Set the value of this variable according to your needs.

Table 221: Voice QoS Behavior

Status	Description
enable	The voice QoS configuration (<i>qosVoiceDiffServ</i> and <i>qosVoiceIeee8021qUserPriority</i> variables) is also applied to the RTCP packets.
disable	The RTCP packets are not tagged by the <i>qosVoiceDiffServ</i> and <i>qosVoiceIeee8021qUserPriority</i> variables.

VLAN

You can set various VLAN parameters to control user priority.

You can also set these parameters via the web interface, as described in ["802.1q Configuration" on page 117](#).

VLANs

VLANs are created with standard Layer 2 Ethernet. A VLAN Identifier (VID) is associated with each VLAN. VLANs offer the following benefits:

- VLANs are supported over all IEEE 802 LAN MAC protocols, and over shared media LANs as well as point-to-point LANs.
- VLANs facilitate easy administration of logical groups of stations that can communicate as if they were on the same LAN. They also facilitate easier administration of moves, adds, and changes in members of these groups.
- Traffic between VLANs is restricted. Bridges forward unicast, multicast, and broadcast traffic only on LAN segments that serve the VLAN to which the traffic belongs.

The VLAN field in the Ethernet file is located after both destination and source addresses:

```

0                               1
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7   (byte)
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| Dest Addr | Src Addr | VLAN | Type/Length | ...
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

The VLAN field is separated as follows:

```

0 (bit)           1           2           3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                               0x8100 | Pri | T |                               VID |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

For both signalling and media packets, the VLAN priority section is configurable independently.

► **To enable the VLAN configuration:**

1. In the *qosVlanleeee8021q* group of the *qosMIB*, locate the *qosVlanleeee8021qTaggingEnable* variable.
2. Set the value of this parameter to **enable**.
The VLAN configuration is enabled.
3. Locate the following variables:
 - *qosVlanleeee8021qVirtualLanID*
 - *qosVlanleeee8021qDefaultUserPriority*
4. Set the value of these variables.
5. Restart the Mediatrix 4100 so that the changes may take effect.

For more information, please refer to the *MIB Reference* manual.

This chapter describes how to configure and use the Syslog daemon.

Syslog Daemon Configuration

Standards Supported

RFC 3164 – The BSD Syslog Protocol

The Syslog daemon is a general purpose utility for monitoring applications and network devices with the TCP/IP protocol. With this software, you can monitor useful messages coming from the Mediatrix 4100 unit. If no Syslog daemon address is provided by a DHCP server or specified by the administrator, no messages are sent.

For instance, if you want to download a new software into the Mediatrix 4100, you can monitor each step of the software download phase. Furthermore, if the unit encounters an abnormal behaviour, you may see accurate messages that will help you troubleshoot the problem.



In the *Unit Manager Network Administration Manual*, refer to chapter *Administration Parameters*, section *Syslog Daemon*.

► To enable the Syslog daemon:

1. In the *syslogMIB*, locate the *syslogMsgMaxSeverity* variable.

This variable indicates which syslog message is processed. Any syslog message with a severity value greater than the selected value is ignored by the agent.

- disabled
- critical
- error
- warning
- informational
- debug

A higher level mask includes lower level masks, e.g., *Warning* includes *Error* and *Critical*. The default value is **informational**.

The following are some of the messages the unit sends:

Table 222: Syslog Messages Examples

Event	Level	Message
The configuration update with the specific configuration file has been successful (configuration file fetching)	Informational	The specific configuration update succeeded.
The configuration update with the specific configuration file experienced an error and has not been completed (configuration file fetching)	Error	The specific configuration update failed.
The software update has been successful	Informational	The software update succeeded.
The software update experienced an error and has not been completed	Error	The software update failed.

Configuration Source

The Mediatrix 4100 must know the IP address and port number of the Syslog server. You can assign these information to the Mediatrix 4100 through a DHCP server or manually enter them yourself with the static variables.

You can also set these parameters via the web interface, as described in [“Syslog Monitoring” on page 29](#).

DHCP Configuration

Using the DHCP configuration assumes that you have properly set your DHCP server with the relevant information. See [“Chapter 8 - IP Address and Network Configuration” on page 143](#) for more details.

► To use DHCP-assigned information:

1. In the *ipAddressConfig* folder, locate the *syslogSelectConfig Source* variable.
This variable defines whether the Mediatrix 4100 shall ask for its Syslog daemon settings through a DHCP server or not.
2. Set the *syslogSelectConfigSource* variable to **dhcp**.
You can query the Syslog daemon's IP address and port number assigned by the DHCP server in the *syslogHost* and *syslogPort* read-only variables (under the *ipAddressStatus Syslog* group of the *ipAddressStatus* folder).
3. Set how you want to define the Syslog information in the DHCP server:

Table 223: Syslog DHCP Information

To use a...	Set...
vendor specific code	The <i>syslogDhcpSiteSpecificCode</i> variable (under the <i>ipAddressConfigSyslogDhcp</i> group) to 0 . Set the Syslog server IP address in the DHCP server inside the vendor specific sub-option 110 (hexadecimal 0x6E).
site specific code	The <i>syslogDhcpSiteSpecificCode</i> variable (under the <i>ipAddressConfigSyslogDhcp</i> group) to any value between 128 and 254. Set the Syslog server IP address in the DHCP server inside the site specific option you have chosen (it must match the value of the <i>syslogDhcpSiteSpecific Code</i> variable in the unit's configuration).

See [“Vendor and Site Specific DHCP Options” on page 156](#) for more details.

Static Configuration

Use the static configuration if you are not using a DHCP server or if you want to bypass it.

► To use static information:

1. In the *ipAddressConfig* folder, locate the *syslogSelectConfig Source* variable.
This variable defines whether the Mediatrix 4100 shall ask for its Syslog daemon settings through a DHCP server or not.
2. Set the *syslogSelectConfigSource* variable to **static**.
3. Set the following variables:

Table 224: Syslog Daemon Static Address

Variable	Description
<i>syslogStaticHost</i>	Syslog server static IP address or domain name. Default Value: 192.168.0.10

Table 224: Syslog Daemon Static Address (Continued)

Variable	Description
syslogStaticPort	Syslog server static IP port number. Default Value: 514

Customizing Syslog Messages

You can display additional information in the prefix of syslog messages the Mediatix 4100 sends. This allows you to later filter the messages. The following is the additional information you can enable:

- ▶ MAC address
- ▶ local time
- ▶ local host



Note: This applies only to syslog messages sent on the network and not the local syslog messages.

▶ **To add the MAC address of the unit in the syslog messages:**

1. In the *syslogMIB*, set the *syslogMsgDisplayMacAddress* variable to **enable**.
The MAC address of the Mediatix 4100 is part of the prefix for all syslog messages.
If you set the variable to **disable**, the MAC address is not displayed in the prefix of the syslog messages.

▶ **To add the local time of the unit in the syslog messages:**

1. In the *syslogMIB*, set the *syslogMsgDisplayTime* variable to **enable**.
The current local time of the Mediatix 4100 is part of the prefix for all syslog messages.
If you set the variable to **disable**, the time is not displayed in the prefix of the syslog messages.

▶ **To add the local host of the unit in the syslog messages:**

1. In the *syslogMIB*, set the *syslogMsgDisplayLocalHost* variable to **enable**.
The current local host of the Mediatix 4100 is part of the prefix for all syslog messages.
If you set the variable to **disable**, the local host is not displayed in the prefix of the syslog messages.

Configuring the Syslog Daemon Application

You shall configure the Syslog daemon to capture those messages. Refer to your Syslog daemon's documentation to learn how to properly configure it to capture messages.

The Mediatrix 4100 collects meaningful statistics that can be read via the RTP MIB.

RTP Statistics

RTP statistics are related to the transmission of information and include, but are not limited to:

- ▶ Number of octets transmitted/received
- ▶ Number of packets transmitted/received
- ▶ Number of lost packets
- ▶ Percentage of lost packets
- ▶ Minimum, maximum and average Jitter interarrival time (time, in milliseconds, between the arrival of packets)
- ▶ Minimum, maximum and average latency time

These statistics are located under the *rtpStats* group of the *rtpMIB*. See the *MIB Reference* manual for more details.

Statistics Buffers

Each statistics has three different buffers in which they are collected:

Table 225: Statistics Buffers

Statistic	Description
Last connection	These are the statistics of the last completed connection.
Current	These are the statistics of the current connection. If using the Cumulated buffer, they are added to the cumulated statistics buffer and then reset.
Cumulated	These are the cumulated statistics of all the connections. Define a period of time and maximum number of periods you want to keep. For instance, you could define to keep the statistics for the last 24 periods of 1 hour.

How are Statistics Collected?

When collecting statistics, you can do so in two ways:

- ▶ Continuous collection of statistics.
In this case, the cumulated statistics are not used (disabled) and the current statistics are constantly updated.
- ▶ Collection of statistics for a defined period of time with a user-defined accuracy.
For instance, you could define to keep the statistics for the last 24 periods of 1 hour.

▶ To set statistics collection:

1. In the *sysConfigMIB*, locate the *sysConfigStats* group.
2. Set the period length you want to keep in the *sysConfigStatsPeriodLength* variable.
The length of a period may vary from 5 minutes to 24 hours, by 5-minutes sections. At expiration, the current statistics are added to the cumulated statistics buffer and then reset. Note that modifying the value of this variable resets statistics to 0.

3. Set the maximum number of periods to cumulate in the *sysConfigStatsNumberPeriods* variable. The maximum number of periods cumulated is 24. If this variable is set to 0, statistics are collected indefinitely in the current variables. Note that modifying the value of this variable resets statistics to 0.

► **To reset statistics:**

1. In the *sysAdminMIB*, set the *sysAdminCommand* variable to **resetStats**. This resets all cumulated call statistics.

Statistics by Syslog

You can configure the Mediatrix 4100 to send the RTP and T.38 statistics by syslog message. You will thus be able to see them by using your syslog daemon.

- The RTP statistics are sent at the end of a call.
- The T.38 statistics are sent at the end of a fax.

The syslog message level is “informational” and uses the module name “Statistics”. [Table 226](#) lists the different statistics fields to send.

Table 226: Statistics by Syslog

Short Name	Description	Corresponding MIB Variable
TxByte	Number of octets transmitted.	rtpStatsLastConnNumberOctetsTransmitted
RxByte	Number of octets received.	rtpStatsLastConnNumberOctetsReceived
TxPkt	Number of packets transmitted.	rtpStatsLastConnNumberPacketsTransmitted
RxPkt	Number of packets received.	rtpStatsLastConnNumberPacketsReceived
NbrPktLost	Number of packets lost.	rtpStatsLastConnNumberPacketsLost
PctPktLost	Percentage of packets lost.	rtpStatsLastConnPercentPacketsLost
JitMin	Minimum interarrival time in milliseconds.	rtpStatsLastConnInterarrivalJitterMin
JitMax	Maximum interarrival time in milliseconds.	rtpStatsLastConnInterarrivalJitterMax
JitAvg	Average interarrival time in milliseconds.	rtpStatsLastConnInterarrivalJitterAvg
LatMin	Minimum latency in milliseconds.	rtpStatsLastConnLatencyMin
LatMax	Maximum latency in milliseconds.	rtpStatsLastConnLatencyMax
LatAvg	Average latency in milliseconds.	rtpStatsLastConnLatencyAvg

The syslog message sent will have the following format:

```
RTP TxByte:<TxByte>, RxByte:<RxByte>, TxPkt:<TxPkt>, RxPkt:<RxPkt>,
NbrPktLost:<NbrPktLost>, PctPktLost:<PctPktLost>, JitMin:<JitMin>,
JitMax:<JitMax>, JitAvg:<JitAvg>, LatMin:<LatMin>, LatMax:<LatMax>,
LatAvg:<LatAvg>
```

Example with the syslog message prefix:

```
Dec 31 19:15:05 10.2.130.31 Statistics [0073] RTP TxByte:32002, RxByte:24514,
TxPkt:156, RxPkt:140, NbrPktLost:0, PctPktLost:0, JitMin:0, JitMax:6, JitAvg:3,
LatMin:8, LatMax:8, LatAvg:8
```

► **To enable to send statistics by syslog:**

1. In the *sysConfigMIB*, set the *sysConfigStatsBySyslogEnable* variable to **enable**.

Example

The following is an example with `sysConfigStatsNumberPeriods = 3` and `sysConfigStatsPeriodLength = 1` (5 minutes).

Table 227: Statistics Setting Example

Statistics	5-minutes sections					
	1	2	3	4	5	6
<code>rtpStatsCurrentTotalOctetsTransmitted</code>	50	30	60	40	100	50
<code>rtpStatsCumulatedTotalOctetsTransmitted</code>	0	50	80	140	130	200

1. 50 total octets transmitted in the first 5-minutes period.
2. 30 total octets transmitted in the second 5-minutes period. The previous statistics are transferred to the corresponding cumulated statistics variable for a cumulated total octets transmitted of 50.
3. 60 total octets transmitted in the third 5-minutes period. The previous statistics are transferred to the corresponding cumulated statistics variable for a cumulated total octets transmitted of 80.
4. 40 total octets transmitted in the fourth 5-minutes period. The previous statistics are transferred to the corresponding cumulated statistics variable for a cumulated total octets transmitted of 140.
5. 100 total octets transmitted in the fifth 5-minutes period. The previous statistics are transferred to the corresponding cumulated statistics variable.

In the above example, the `rtpStatsCumulatedxx` variables always contain the statistics for the last 15 minutes (`sysConfigStatsNumberPeriods X sysConfigStatsPeriodLength`) accurate to 5 minutes (`sysConfigStatsPeriodLength`). This means that the statistics for the first 5-minutes period are dropped, for a cumulated total octets transmitted of 130.

6. 50 total octets transmitted in the sixth 5-minutes period. The previous statistics are transferred to the corresponding cumulated statistics variable.
The statistics for the second 5-minutes period are dropped, for a cumulated total octets transmitted of 200.

Maximum Transmission Unit (MTU)

This chapter describes the MTU (Maximum Transmission Unit) requirements of the Mediatrix 4100.

What is MTU?

The *Maximum Transmission Unit* (MTU) is a parameter that determines the largest packet than can be transmitted by an IP interface (without it needing to be broken down into smaller units). Each interface used by TCP/IP may have a different MTU value specified.

The MTU should be larger than or equal to the largest packet you wish to transmit unfragmented. Note that this only prevents fragmentation locally. Some other link in the path may have a smaller MTU: the packet will be fragmented at that point, although some routers may refuse packets larger than their MTU.

Mediatrix 4100 MTU

The Mediatrix 4100 MTU is 1500 bytes, which is the Ethernet typical value.

Possible Hardware Problem

The implementation of the IEEE Standard 802.1q in the Mediatrix 4100 may have a minor problem because of hardware limitations.

802.1q increases the Ethernet frame header by 4 bytes, adding a Virtual LAN ID and a user_priority. This is useful to limit broadcasts that cross bridges, and it may also prioritize frames in the queuing algorithm of switches. However, it also increases the maximum possible size of Ethernet frames from 1518 to 1522 bytes, and this might not be handled adequately by every hardware.

A workaround is available for PCs running Windows to avoid sending 1522 bytes packets (note that this happens only in special and rare cases). The workaround is to reduce the MTU of the interface (the one that sends packets with 802.1q framing) by 4 bytes.

1. Use the registry editor (regedt32) and go to the key:
Windows 2000 and later:
`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\Interfaces\<ethernet adapter>`
Windows NT4 and 98:
`\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\<ethernet adapter>\Parameters\Tcpip`
where <Ethernet adapter> can be found by using the command "ipconfig /all".
2. Add (or modify) a value named MTU of type REG_DWORD. Set it to 1496 (instead of 1500), in decimal. Restart the computer to have those changes in effect.
In Windows 2000 and later this value is under the following key:
 - Key: `Tcpip\Parameters\Interfaces\ID` for Adapter2

- Value Type: REG_DWORD Number
 - Valid Range: 68 - the MTU of the underlying network
 - Default: 0xFFFFFFFF
 - Description: This parameter overrides the default MTU for a network interface. The MTU is the maximum packet size in bytes that the transport will transmit over the underlying network. The size includes the transport header. Note that an IP datagram may span multiple packets. Values larger than the default for the underlying network will result in the transport using the network default MTU. Values smaller than 68 will result in the transport using an MTU of 68.
3. To validate that the changes are correct, try to ping the Mediatrix 4100 with large packets once restarted:
`ping -l 2000`
- This will cause IP fragmentation, the first fragment being as large as the interface allows it. With the MTU reduced, you should now receive an answer. For more informations, see:
<http://support.microsoft.com/default.aspx?scid=kb;en-us;120642>.

You can experience some problems when connecting the Mediatrix 4100 to the network. The following section examines some of these problems and possible solutions.

A Syslog message lists the problems the Mediatrix 4100 encounters. You can see this message with the Syslog daemon.

This chapter covers the following types of issues:

- ▶ General Operation Issues
- ▶ Calling Issues
- ▶ Fax Issues
- ▶ Configuration Issues
- ▶ Software Upgrade Issues
- ▶ SNMP Management Software Issues

General Operation Issues

The following are general operation issues you may encounter.

DESCRIPTION: Unit does not operate – All LEDs are OFF.

■ **POSSIBLE CAUSE:** Power is not fed to the unit.

SOLUTION: Check that:

- The power cord is connected to the electrical outlet.
- The power cord is fully inserted into the Mediatrix 4100 power socket.

DESCRIPTION: There is a long delay when starting the Mediatrix 4100.

■ **POSSIBLE CAUSE:** If any information is set to come from the DHCP server (for example, SNTP address), the restarting unit waits for a maximum period of two minutes if the DHCP server cannot be reached, even if most other settings are set to “static”.

This delay is caused by the Mediatrix 4100 that cannot function as configured if part of its configuration (the DHCP information) is unavailable.

The two minutes waiting period is an issue with switches that use the Spanning Tree Protocol. When this protocol is enabled, the restarting Mediatrix 4100 may be denied from the network for a certain time (about two minutes). The unit must not ignore transmission errors (i.e., timeouts) because these errors might be caused by the Spanning Tree Protocol.

SOLUTION: Media5 recommends to set up all information to use a static value, or have a DHCP server answer the requests. See [“Static Configuration” on page 145](#) for more details.

DESCRIPTION: I changed the IP address of my unit, but I can't reach the DHCP server anymore.

- **POSSIBLE CAUSE:** A subnet mask is used to determine to which subnet an IP address belongs. An IP address has two components, the network address and the host address. For example, let's consider the IP address 192.168.0.1. Assuming this is part of a Class B network, the first two numbers (192.168) represent the Class B network address, and the second two numbers (0.1) identify a particular host on this network.

Let's say you have the following information:

- Mediatix 4100 IP address: 192.168.0.1
- Subnet Mask: 255.255.0.0 (Class B)
- DHCP Server IP address: 192.168.0.20

If you happen to change the Mediatix 4100 IP address to 192.169.0.1, for instance, the subnet mask is still valid, but cannot reach your DHCP server anymore. Refer to subnet mask documentation for more details.

DESCRIPTION: Unable to reach the Mediatix 4100 after changing the Ethernet speed at run-time.

- **POSSIBLE CAUSE:** Some hubs cannot adapt completely their port speed at run-time.

SOLUTION: Always restart the Mediatix 4100 for the new setting to take effect. See ["Ethernet Connection Speed" on page 162](#) for more details.

DESCRIPTION: The Bypass feature does not activate if the SIP proxy times out when a call is initiated.

- **POSSIBLE CAUSE:** In SIP, there is no direct correlation between the user agent and the proxy. The user agent may be able to complete outgoing calls without the help of the server, and may also receive calls as well. The problem is thus normal. The SIP proxy going down is rather a network setup problem.

SOLUTION: To avoid those types of failures, the network should use redundant servers when possible.

DESCRIPTION: Setting the MIB variable *voicelfAdaptativeJitterBufferEnable* to **disable** has no effect.

- **POSSIBLE CAUSE:** You cannot disable the adaptative jitter buffer on the Mediatix 4100.

SOLUTION: If you set the *voicelfTargetJitterBufferLength* and *voicelfMaxJitterBufferLength* variables to the same value, you will have a non-adaptative jitter buffer. See ["Adaptative Jitter Buffer" on page 243](#) for more details.

DESCRIPTION: When I set values such as the User Name and Display Name, the value is not accepted and is reset to its default value once the Mediatix 4100 restarts.

- **POSSIBLE CAUSE:** When you enter values that contain non-standard English characters in entries that accept strings of characters, this invalidates the value and resets it to its default value. However, this may be visible only once the Mediatix 4100 restarts.

SOLUTION: Make sure that your string of characters contain only characters that are part of the following ASCII characters list:

10	LF, line feed	62	>, greater than	94	^, caret
13	CR, carriage return	63	?, question mark	95	_, underscore
32	space	64	@, commercial at	96	`, back quote
33	!, exclamation mark	65	A	97	a
34	", double quote	66	B	98	b
35	#, hash	67	C	99	c
36	\$, dollar	68	D	100	d
37	%, percent	69	E	101	e
38	&, ampersand	70	F	102	f
39	', quote	71	G	103	g
40	(, open parenthesis	72	H	104	h
41	), close parenthesis	73	I	105	i
42	*, asterisk	74	J	106	j
43	+, plus	75	K	107	k
44	,, comma	76	L	108	l
45	-, minus	77	M	109	m
46	., full stop	78	N	110	n
47	/, oblique stroke	79	O	111	o
48	0, zero	80	P	112	p
49	1	81	Q	113	q
50	2	82	R	114	r
51	3	83	S	115	s
52	4	84	T	116	t
53	5	85	U	117	u
54	6	86	V	118	v
55	7	87	W	119	w
56	8	88	X	120	x
57	9	89	Y	121	y
58	:, colon	90	Z	122	z
59	;, semicolon	91	[, open square bracket	123	{, open curly bracket
60	<, less than	92	\, backslash	124	, vertical bar
61	=, equals	93	], close square bracket	125	}, close curly bracket
				126	~, tilde

DESCRIPTION: Media5 Technical Support personnel asked me to enable the PCM traces. How do I do it?



POSSIBLE CAUSE: PCM traces are an efficient tool to identify problems with:

- Echo in your network
- DTMF signals
- Caller ID signals
- Fax signals (or false Fax detection)
- Message Waiting Indicator signals
- Any other analog signal

SOLUTION: Do the following:

- Enable the PCM traces by setting the *mxDebugPcmCaptureEnable* MIB variable to **enable**.
- Set the destination IP address for the PCM traces in the *mxDebugPcmCaptureIpAddress* MIB variable.
 - This IP address does not have to be listening on ports 5001/2 - 6001/2, as it is easy to filter out ICMP "port unreachable" messages afterwards.
 - The PCM traces destination must be set so it can be recorded in a Wireshark capture on your network, normally sent to the PC doing the capture.
- Set the endpoint number on which to perform the PCM capture in the *mxDebugPcmCaptureEndpointNumber* variable.

For more details on the PCM traces, refer to *Technical Bulletin 0648 - PCM Traces*.

Calling Issues

The following are general calling issues you may encounter.

DESCRIPTION: Impossible to make a call.

If the following happens:

- ▶ Dial tone present.
- ▶ *Power* LED lit.
- ▶ *LAN* LED lit.

■ **POSSIBLE CAUSE:** Network communication is not working.

SOLUTION: Check that:

- The LAN cable is securely connected to the Mediatrix 4100 and to the network connector.
- You did not connect a crossover network cable.

■ **POSSIBLE CAUSE:** Configurable parameters of the Mediatrix 4100 are not set properly.

SOLUTION: Refer to this manual for a complete description of the configurable Mediatrix 4100 parameters.

DESCRIPTION: Cannot make or receive calls.

■ **POSSIBLE CAUSE:** There may be calls that have not been properly terminated, which causes a “leak” in the system.

SOLUTION: You can enable the SIP Context Snapshot time feature. This feature is used to find if there are improperly terminated calls. This could help to debug the system.

- a. In the *syslogMIB*, set the *syslogMsgMaxSeverity* variable to **debug**.
- b. Configure and enable the syslog feature.
- c. In the *sipDebugMIB*, set the time, in minutes, between snapshots in the *sipDebugContextSnapshotTime* variable.

The list of contexts currently in use are periodically output as debug-level syslog messages. Note that enabling this feature will also trigger an instant snapshot.

To disable the feature, set this variable to zero (0).

Note that this feature will generate more syslog traffic, about 20 messages at each x minutes.



Note: This feature is currently located under the *mediatrixExperimental* branch of the MIB structure. See [“MIB Structure” on page 135](#) for more details.

■ **POSSIBLE CAUSE:** It is possible that the unit is refreshing its registration and has entered a race condition between the refresh and the SIP timeouts. Normally, the Mediatrix 4100 cannot make or receive calls until the REGISTER request has completed successfully. Because the timeout for a SIP transaction in UDP is 32 seconds, it is possible to have an ongoing re-REGISTER transaction at the same moment that the registration itself expires. This could happen if the *sipReRegistrationTime* variable is set to a value lower than 32. In that case, the user agent becomes unregistered, and will become registered again only when the re-REGISTER request is answered with a positive response from the server.

SOLUTION: Set the *sipUnregisteredPortBehavior* variable to **enablePort**. This way, when an endpoint is not registered, it is still enabled. The user can receive and initiate calls. See “Unregistered Line Behaviour” on page 154 for more details. See also [“Refreshing Registration” on page 268](#) for more details on the re-registration feature.

DESCRIPTION: When making a 3-way conference, part of the conversation is lost, resulting in a choppy voice.

- **POSSIBLE CAUSE:** The packetization period (ptime) is not the same for all the participants of the conference, which causes the choppy voice issue.

SOLUTION: For better results, Media5 recommends to set the packetization period of all participants of a 3-way conference to 30 milliseconds. See [“Packetization Time” on page 234](#) for more information on how to set the packetization period of the Mediatrix 4100.

DESCRIPTION: Unable to establish a call from the Mediatrix 4100 to a user agent such as an IP phone, a gateway or another access device.

- **POSSIBLE CAUSE:** When the Mediatrix 4100 – with its T.38 capability enabled – tries to establish a call with a user agent that does not support T.38, this a user agent rejects the call instead of ignoring the capability it does not support, i.e., T.38.

SOLUTION: Disable the T.38 capability in the Mediatrix 4100. See [“T.38 Fax” on page 254](#) for more details.

Fax Issues

The following gives information pertaining to faxes. This includes a list of fax models tested with the Mediatrix 4100 and some specific issues the unit may encounter.

DESCRIPTION: “Poor line condition” error during a fax transmission.

- **POSSIBLE CAUSE:** The analog transmission between the fax machine and the Mediatrix 4100 is flaky, preventing the fax transmission to terminate properly. This problem is known to occur with some fax machines and it can also occur with a few fax modems.

SOLUTION: Set the *Input sound level* to **-6 dB**. If this still does not solve the problem, try the **+6 dB** value. See [“User Gain” on page 246](#) for more details.

DESCRIPTION: Unable to send a fax in T.38 and Clear Channel.

- **POSSIBLE CAUSE:** To properly send faxes, both units must be configured with the same settings. If you are attempting to send a fax and the transmission fails, there could be many reasons for this, but most likely the fax codec settings are at fault. The following explains the logic behind fax transmissions.

When transmitting a fax, Unit A first verifies if Unit B supports the codec you have set in Unit A. If the codec is supported, the fax should be transmitted properly.

If the fax codec is not supported by Unit B, Unit A tries to find a common preferred G.711 clear channel codec between the two units. If Unit A finds one, it uses this common clear channel codec and the fax should be transmitted properly. If there are no common clear channel codecs between the units, the fax transmission fails.

SOLUTION: To avoid fax transmission problems, configure both units with the same T.38 and clear channel settings and the fax should be sent properly.

DESCRIPTION: The T.38 fax transmission fails.

- **POSSIBLE CAUSE:** The Mediatrix 4100 opens the T.38 channel only after receiving the “200 OK” message from the peer. This means that the Mediatrix 4100 cannot receive T.38 packets before receiving the “200 OK”. Based on RFC 3264, the T.38 channel should be opened as soon as the unit sends the “INVITE” message.

Information from RFC 3264 (An Offer/Answer Model with Session Description Protocol (SDP)) - section 5.1: Once the offerer has sent the offer, it must be prepared to receive media for any recvonly streams described by that offer. It must be prepared to send and receive media for any sendrecv streams in the offer, and send media for any sendonly streams in the offer (of course, it cannot actually send until the peer provides an answer with the needed address and port information). In the case of RTP, even though it may receive media before the answer arrives, it cannot send RTCP receiver reports until the answer arrives.

SOLUTION: Be sure to reply to the “INVITE” message by a “200 OK” before sending any T.38 message to the Mediatrix 4100.

DESCRIPTION: Voice does not switch back to the original negotiated codec after a clear channel fax is performed.

- **POSSIBLE CAUSE:** The Mediatrix 4100 suffers from a limitation of its DSP. The Mediatrix 4100 cannot detect the end of a clear channel fax, which means that the unit cannot switch back to the original negotiated codec if this codec was not a clear channel codec, e.g., a session established in G.729.

When the unit detects a fax, it automatically switches to a negotiated clear channel codec such as PCMU (if there is no T.38 or if T.38 negotiation failed). Once the fax is terminated, the Mediatrix 4100 is not notified by the DSP. The unit thus stays in the clear channel codec and does not switch back to G.729.

SOLUTION: There is no solution.

Tested Fax Models

The following table lists the fax models tested with the Mediatrix 4100 for the T.38 protocol. Each of these fax models has been emulated and tested with each other by using the FaxLab[®] fax/telephony testing tool.

Table 228: Tested Fax Models

Make	Models	
Brother	• 6650MC • 7150C • FAX-190 • FAX-580MC • Intellifax 600 • Intellifax 625	• Intellifax 950M • Intellifax 2500 • MFC 4550 • MFC 4600 • MFC 4650
Panasonic	• KXF-500 • KXF-580 • KXF-1600 • KXF-3000 • KX-FP270 • KX-FPC95	• PX-5 • PX-150 • PX-350 • UF-880 • UF-V60
Sharp	• FO-145 • FO-235 • FO-445 • FO-5400 • UX-104 • UX-108	• UX-117 • UX-256 • UX-460 • UX-1400 • UX-3600M

Table 228: Tested Fax Models (Continued)

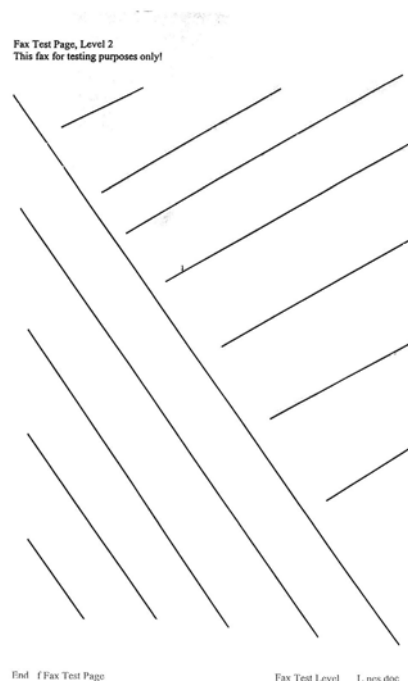
Make	Models	
Canon	• B70 • Fax 750 • Fax B340 • L777 • MultiPass C530 • MultiPass C545 • MultiPass C555	• MultiPass C560 • MultiPass C755 • MultiPass C2500 • MultiPass C5500 • MultiPass L6000 • MultiPass TF-301
Xerox	• 3004 • 7021 • 7024 • 7033 • WorkCenter 250	• WorkCenter 470cx • WorkCenter 480cx • WorkCenter XE90fx • WorkCenter XK50cx
Hewlett Packard	• Fax-200 • Fax 920 • LaserJet 3200	• OfficeJet • OfficeJet 350 • OfficeJet 570

Issues Arising from Specific Combinations/Scenarios

The following are very specific issues the Mediatrix 4100 may experience with certain types of faxes.

DESCRIPTION: Fax in T.38 fails with particular fax machines and fax test sheet.

■ **ISSUE:** Faxes fail when using Brother 255 Model faxes while faxing specific test sheet. The following is the test sheet used:

Figure 93: Test Sheet Used

DESCRIPTION: Fax in T.38 may fail between a Mediatrix 1104 and a Mediatrix 1124 when using a Canon Multipass C530 and Panasonic PX-5.

- **ISSUE:** The specific fax combination of Canon MultiPass C530 and Panasonic PX-5 does not work with the following settings:
- Originating fax: Canon Multipass C530, non-ECM, best encoding, V.17, 14400 bps.
 - Terminating fax: Panasonic PX-5, best error correction, best encoding, V.29, 9600 bps.
-

DESCRIPTION: When using the Mediatrix 4100 with the Cyberguard SG530 broadband router, the router blocks fax transmissions.

- **POSSIBLE CAUSE:** Cyberguard Version 2.0.2 seems to be the problem.
SOLUTION: Upgrade the Cyberguard to version 2.1.3.
-

Configuration Issues

The following are issues you may encounter when changing the Mediatrix 4100 configuration.

DESCRIPTION: When the Mediatrix 4100 configuration is entirely static and I change the configuration source of any server from static to DHCP, the service related to the server is not accessible.

- **POSSIBLE CAUSE:** If none of the *xxxConfigSource* variables (in the *ipAddressStatus* folder) are set to **dhcp**, then the Mediatrix 4100 does not send a DHCP REQUEST message. This is the case if:
- you set all *xxxSelectConfigSource* variables to something other than **dhcp** and you restart the Mediatrix 4100, or
 - you select the **setConfigSourcesStatic** option of the *sysAdminCommand* variable and you restart the Mediatrix 4100.

Whenever the *xxxSelectConfigSource* variable of a specific server, e.g., syslog server, is set to **dhcp**, then no IP address can be assigned to that server (this does not trigger a DHCP request).

The service is therefore not functional, the corresponding *xxxHost* variable is set to **0.0.0.0**, and the corresponding *xxxPort* variable is not accessible (the GET request result is ERROR), in the *ipAddressStatus* folder.



Note: In the case of the SIP servers, the corresponding *xxxPort* variable is accessible.

SOLUTION: Restart the Mediatrix 4100 or set the proper *xxxSelectConfigSource* variable to **static**.

Software Upgrade Issues

The following are issues you may encounter when performing a software upgrade operation.

DESCRIPTION: An error occurs when the Mediatrix 4100 attempts to communicate with the image server.

- **POSSIBLE CAUSE:** The directory specified in the upgrade command does not exist or does not contain the files required for the software download process.
-

SOLUTION:

- Check the directory name.
- Be sure that the directory contains files. If not, extract them from the zip file again. See ["Download Procedure" on page 211](#) for more details.
- Be sure that the software server is running and properly configured.

■ **POSSIBLE CAUSE:** The IP address of the software server is not the correct one.

SOLUTION:

- Check the given IP address.
- Check the IP port.

DESCRIPTION: An error occurs when the Mediatrix 4100 attempts to transfer the software upgrade.

■ **POSSIBLE CAUSE:** The Ethernet cable has become disconnected from the Mediatrix 4100 or the PC running the file transfer.

SOLUTION: Reconnect the cable and start again.

■ **POSSIBLE CAUSE:** Power to the Mediatrix 4100 has been disrupted during the file transfer.

SOLUTION: Check the power connection to the Mediatrix 4100 and start again.

DESCRIPTION: When downgrading the Mediatrix 4100 to a previous version of the application software, the unit does not restart, the LANLED is blinking and all other LEDs are off.

■ **POSSIBLE CAUSE:** The default router IP address is set to 0.0.0.0, which is not supported by the version to which you downgraded.

SOLUTION: Perform a recovery mode or a factory reset procedure after proceeding with the downgrade operation.

- If you perform a recovery mode as per ["Recovery Mode" on page 20](#), you must manually change the default router IP address to a valid address other than 0.0.0.0, then restart the Mediatrix 4100.
- If you perform a factory reset procedure as per ["Factory Reset" on page 21](#), everything should be working properly. However, this deletes any custom setting you may have done in other variables as it reverts the Mediatrix 4100 back to its default factory settings.

DESCRIPTION: The TFTP server does not recognize the download path and produces an error.

■ **POSSIBLE CAUSE:** You should use the "/" character when defining the path to indicate sub-directories, i.e., *c:/temp/download*. However, some TFTP servers on the Windows operating system do not recognize the "/" character and produce an error.

SOLUTION: Use the "\" character in the path definition.

DESCRIPTION: Performing a software download takes an unusually long time.

■ **POSSIBLE CAUSE:** If the following happens:

- Any information is set to come from the DHCP server (for example, the SNTP server address) and the DHCP server cannot be reached.
- The primary software server address is invalid (either set by DHCP or static).

The unit tries to reach the primary software server without realizing that the address is invalid. It keeps trying for a few minutes, even if the download procedure fails.

This delay is caused by the Mediatrix 4100 that cannot function as configured if part of its configuration (the DHCP information) is unavailable. Furthermore, there is an issue with switches that use the Spanning Tree Protocol. When this protocol is enabled, the Mediatrix 4100 may be denied from the network for a certain time, which causes the long delay.

SOLUTION: Media5 recommends to set up all information to use a valid static value, or have a DHCP server answer the requests. See ["Static Configuration" on page 145](#) for more details.

SNMP Management Software Issues

The following are issues you may encounter when trying to contact the Mediatrix 4100 with a SNMP management software.

DESCRIPTION: The SNMP network management software cannot access the Mediatrix 4100.

■ **POSSIBLE CAUSE:** The SNMP network management software does not have the proper Mediatrix 4100 information.

SOLUTION: Check that:

- The IP information for the Mediatrix 4100 is correctly configured.
- The Mediatrix 4100 was restarted after defining the IP information.
- The line through which you are trying to access the Mediatrix 4100 has been unlocked or is not the correct line. If it is locked, check the connections and network cabling for the connector.

Try to locate the Mediatrix 4100 IP address. If impossible, perform a recovery reset as indicated in section ["Reset / Default Switch" on page 19](#).

DESCRIPTION: There is no response when trying to access the Mediatrix 4100.

■ **POSSIBLE CAUSE:** The Mediatrix 4100 speaks the three most common SNMP protocols: SNMPv1, SNMPv2c, and SNMPv3. If you try to access it by using any other protocol, it stays silent.

DESCRIPTION: The SNMP network manager does not receive Traps.

■ **POSSIBLE CAUSE:** The IP information is not correct.

SOLUTION: Check that the IP information (IP address + IP port) of the SNMP network manager software is correctly recorded by the Mediatrix 4100.

DESCRIPTION: When trying to set a variable, the Mediatrix 4100 does no respond, nor sends an error message.

■ **POSSIBLE CAUSE:** In secure management mode, the Mediatrix 4100 does not accept SNMPv1 and SNMPv2c SET requests. However, the MIB variables are viewable in any management mode (secure and not secure).

DESCRIPTION: When entering a value such as “.23” in a MIB variable (for instance, *sipTransportQValue*), the Mediatrix 4100 returns a “Wrong value” error message.

- **POSSIBLE CAUSE:** The Mediatrix 4100 does not support a value such as “.23”.
- **SOLUTION:** Enter a value such as “0.23” instead.

DESCRIPTION: When I try to set a variable with a MIB configuration tool such as Media5 Unit Manager Network, nothing happens.

- **POSSIBLE CAUSE:** The variable may be in a MIB that is located under the *mediatrixExperimental* branch of the MIB structure.
Media5 configuration tool – the Unit Manager Network – does not support MIBs that are located under the *mediatrixExperimental* branch of the MIB structure. The Unit Manager Network does not have specific tasks to manage variables in experimental MIBs.
The *mediatrixExperimental* branch is the area where objects and events in MIBs under development can be placed without fear of conflicting with other MIBs. When the items rooted under an experimental sub-tree are ready for release, they will be under a permanent branch.
Even though the Unit Manager Network can view experimental MIBs, SNMP operations may not work properly on them.

DESCRIPTION: When viewing a table, the unit does not respond.

- **POSSIBLE CAUSE:** It may take time to fill completely a table: from 1 to 5 seconds. This is normal, because the unit is an embedded device with limited processing power.

DESCRIPTION: Is it possible for a hacker to change the content of SNMPv3 variables once the Mediatrix 4100 is in secure mode management?

- **POSSIBLE CAUSE:** In secure management mode, the Mediatrix 4100 works in SNMPv1 read-only, SNMPv2c read-only, and SNMPv3 read/write. SNMP requests using the first two protocols are read-only, and tables used for setting up SNMPv3 users hide the passwords they carry. Because hackers do not know what password to use in SNMPv3 requests, they cannot access the Mediatrix 4100 with read-write permission.

Appendices

Page Left Intentionally Blank

Standards Compliance and Safety Information

This Appendix lists the various standards compliance of the Mediatrix 4100.

Standards Supported

The Mediatrix 4100 complies to the following standards:

Table 229: Standards Compliance

Category	Specification
Agency approvals	- European Union, CE mark (Declaration of Conformity) - Anatel - CSA - NOM - ACME (A-Tick) - Australia - Telepermit - New Zealand - FCC
Safety standards	- UL60950-1 3rd Edition (2000) - CAN/CSA-C22.2 No. 60950-00 - IEC 60950 (1st Edition 2001 With all national deviations) - NOM-019-SCFI-1998 - Anatel Resolution 238:2000
Emissions	- FCC Part 15:2009 Class B - EN55022 (2006) Class B - EN61000-3-2 (1995) Harmonic current emissions - EN61000-3-3 (1995) Voltage fluctuations and flicker - Anatel Resolution 237:2000 (Title II)
Immunity	EN55024:1998 with amendments A1 (2001) (2003) including the following: - EN61000-4-2 (1995), ESD - EN61000-4-3 (1996), Radiated RF - EN61000-4-4 (1995), Burst Transients - EN61000-4-5 (1995), Surge - EN61000-4-6 (1996), Conducted RF - EN61000-4-11 (1995), Voltage Dips and Interruptions

Table 229: Standards Compliance (Continued)

Category	Specification
Telecom	- FCC Part 68:Subpart D - Industry Canada (CS-03, Issue 8, Part 1) - TBR 21: January 1998 - PTC 200: 2006 - PTC 220: 2008 - ITU-K21 (2008) clauses 5.1.1a.b and 5.1.2a.b.



Note: The standards compliance of the Mediatrix 4100 are printed on a sticker located on the bottom of the unit.

Disclaimers

The following are the disclaimers related to the Mediatrix 4100.

Federal Communications Commission (FCC) Part 15

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



Note: Any changes or modifications not expressly approved by Media5 could void the user's authority to operate the equipment.

Federal Communications Commission (FCC) Part 68

This equipment complies with Part 68 of the FCC Rules. On the underside of this equipment is a label that contains, among other information, the FCC Registration Number, Ringer Equivalence Number (REN) and USOC jack type for this equipment. You must, upon request, provide this information to your telephone company.

The REN is useful to determine the quantity of devices you may connect to your telephone line and still have all of those devices ring when your telephone number is called. In most, but not all areas, the sum of the REN's of all devices connected to one line should not exceed five (5.0). To be certain of the number of devices you may connect to your line, as determined by the REN, you should contact your telephone company to determine the maximum REN for your calling area. If your telephone equipment causes harm to the telephone network, the Telephone Company may discontinue your service temporarily. If possible, they will notify you in advance, but if advance notice is not practical, you will be notified as soon as possible. You will be informed of your right to file a complaint with the FCC.

Your telephone company may make changes in its facilities, equipment, operations or procedures that could affect the proper functioning of your equipment. If they do, you will be notified in advance to give you an opportunity to maintain uninterrupted telephone service.

If you experience trouble with this telephone equipment, please contact Media5 for information on how to obtain service or repairs. The telephone company may ask that you disconnect this equipment from the network until the problem has been corrected or until you are sure that the equipment is not malfunctioning.

This equipment may not be used on coin service provided by the telephone company.

Connection to party lines is subject to state tariffs.

INSTALLATION

This device is equipped with an USOC RJ-11C connector.

Industry Canada

This Class B digital apparatus complies with Canadian ICES-003.

Cet appareil numérique de la classe B est conforme à la norme NMB-003 du Canada.

The Industry Canada Label identifies certified equipment. This certification means that the equipment meets telecommunications network protective, operational and safety requirements as prescribed in the appropriate Terminal Equipment Technical Requirements document(s). The Department does not guarantee the equipment will operate to the user's satisfaction.

Before installing this equipment, users should ensure that it is permissible to be connected to the facilities of the local telecommunications company. The equipment must also be installed using an acceptable method of connection. The customer should be aware that compliance with the above conditions may not prevent degradation of service in some situations.

Repairs to certified equipment should be coordinated by a representative designated by the supplier. Any repairs or alterations made by the user to this equipment, or equipment malfunctions, may give the telecommunications company cause to request the user to disconnect the equipment.

Users should ensure for their own protection that the electrical ground connections of the power utility, telephone lines and internal metallic water pipe system, if present, are connected together. This precaution may be particularly important in rural areas.



Warning: Users should not attempt to make such connections themselves, but should contact the appropriate electric inspection authority, or electrician, as appropriate.



Note: The Ringer Equivalence Number (REN) for this terminal equipment is 0.0. The Ringer Equivalence Number (REN) assigned to each terminal device provides an indication of the maximum number of terminals allowed to be connected to a telephone interface. The termination on an interface may consist of any combination of devices subject only to the requirement that the sum of the Ringer Equivalence Number of all the devices does not exceed 5.



Note: This equipment meets the applicable Industry Canada Terminal Equipment Technical Specifications. This is confirmed by the registration number. 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.

CE Marking



DECLARATION OF CONFORMITY

We Media5 Corporation located at 4229 Garlock st. Sherbrooke, Québec, Canada J1L 2C8 declare that for the hereinafter mentioned product the presumption of conformity with the applicable essential requirements of DIRECTIVE 1999/5/EC OF THE EUROPEAN PARLIAMENT (RTTE DIRECTIVE) is given.

Any unauthorized modification of the product voids this declaration.

For a copy of the original signed Declaration Of Conformity please contact Media5 at the above address.

RoHS China

这个文件涉及的是在中华人民共和国境内进口或销售的电子信息产品

Include this document with all Electronic Information Products imported or sold in the People’s Republic of China

部件名称 (Parts)	有毒有害物质或元素 (Hazardous Substance)					
	铅 (Pb)	汞 (Hg)	镉 (Cd)	六价铬 (Cr ⁶⁺)	多溴联苯 (PBB)	多溴二苯醚 (PBDE)
塑料和聚合物部件 (Plastic and Polymeric parts)	○	○	○	○	×	×
集成电路 (Integrated Circuit)	×	○	×	○	×	×
○： 表示该有毒有害物质在该部件所有均质材料中的含量均在 SJ/T-11363 - 2006规定的限量要求以下。 Indicates that the concentration of the hazardous substance in all homogeneous materials in the parts is below the relevant threshold of the SJ/T-11363 - 2006 standard. ×： 表示该有毒有害物质至少在该部件的某一均质材料中的含量可能超出SJ/T-11363 - 2006规定的限量要求。 Indicates that the concentration of the hazardous substance of at least one of all homogeneous materials in the parts might exceed the relevant threshold of the SJ/T-11363 - 2006 standard.						

除非另外特别的标注,此标志为针对所涉及产品的环保使用期限标志. 某些可更换的零部件会有一个不同的环保使用期限(例如,电池单元模块)贴在其产品上.

此环保使用期限只适用于产品是在产品手册中所规定的条件下工作.

The Environmentally Friendly Use Period (EFUP) for all enclosed products and their parts are per the symbol shown here, unless otherwise marked. Certain field-replaceable parts have a different EFUP (for example, battery modules) and so are marked to reflect such. The Environmentally Friendly Use Period is valid only when the product is operated under the conditions defined in the product manual.

Translated Warning Definition

The following information provides an explanation of the symbols which appear on the Mediatrix 4100 and in the documentation for the product.



Warning: Means danger. You are in a situation that could cause bodily injury. Before you work on any equipment, you must be aware of the hazards involved with electrical circuitry and familiar with standard practices for preventing accidents.

Waarschuwing: Dit waarschuwingssymbool betekent gevaar. U overtreedt in een situatie die lichamelijk letsel kan veroorzaken. Voordat u aan enige apparatuur gaat werken, dient u zich bewust te zijn van de bij elektrische schakelingen betrokken risico's en dient u op de hoogte te zijn van standaard maatregelen om ongelukken te voorkomen.

Varoitus: Tämä varoitusmerkki merkitsee vaaraa. Olet tilanteessa, joka voi johtaa ruumiinvammaan. Ennen kuin työskentelet minkään laitteiston parissa, ota selvää sähkökytkentöihin liittyvistä vaaroista ja tavanomaisista onnettomuuksien ehkäisykeinoista.

Attention: Ce symbole d'avertissement indique un danger. Vous vous trouvez dans une situation pouvant causer des blessures ou des dommages corporels. Avant de travailler sur un équipement, soyez conscient des dangers posés par les circuits électriques et familiarisez-vous avec les procédures couramment utilisées pour éviter les accidents.

Warnung: Dieses Warnsymbol bedeutet Gefahr. Sie befinden sich in einer Situation, die zu einer Körperverletzung führen könnte. Bevor Sie mit der Arbeit an irgendeinem Gerät beginnen, seien Sie sich der mit elektrischen Stromkreisen verbundenen Gefahren und der Standardpraktiken zur Vermeidung von Unfällen bewußt.

Avvertenza: Questo simbolo di avvertenza indica un pericolo. La situazione potrebbe causare infortuni alle persone. Prima di lavorare su qualsiasi apparecchiatura, occorre conoscere i pericoli relativi ai circuiti elettrici ed essere al corrente delle pratiche standard per la prevenzione di incidenti.

Advarsel: Dette varselsymbolet betyr fare. Du befinner deg i en situasjon som kan føre til personskade. Før du utfører arbeid på utstyr, må du være oppmerksom på de faremomentene som elektriske kretser innebærer, samt gjøre deg kjent med vanlig praksis når det gjelder å unngå ulykker.

Aviso: Este símbolo de aviso indica perigo. Encontra-se numa situação que lhe poderá causar danos físicos. Antes de começar a trabalhar com qualquer equipamento, familiarize-se com os perigos relacionados com circuitos eléctricos, e com quaisquer práticas comuns que possam prevenir possíveis acidentes.

¡Advertencia!: Este símbolo de aviso significa peligro. Existe riesgo para su integridad física. Antes de manipular cualquier equipo, considerar los riesgos que entraña la corriente eléctrica y familiarizarse con los procedimientos estándar de prevención de accidentes.

Varning!: Denna varningssymbol signalerar fara. Du befinner dig i en situation som kan leda till personskada. Innan du utför arbete på någon utrustning måste du vara medveten om farorna med elkretsar och känna till vanligt förfarande för att förebygga skador.

Safety Warnings

This section lists the following safety warnings:

- ▶ Circuit Breaker (15A) Warning
- ▶ TN Power Warning
- ▶ Product Disposal Warning
- ▶ No. 26 AWG Warning
- ▶ LAN and FXS Ports Connectors Warning
- ▶ Socket Outlet Warning

Circuit Breaker (15A) Warning



Warning: This product relies on the building's installation for short-circuit (overcurrent) protection. Ensure that a fuse or circuit breaker no larger than 120 VAC, 15A U.S. (240 VAC, 10A international) is used on the phase conductors (all current-carrying conductors).

TN Power Warning



Warning: The device is designed to work with TN power systems.

Product Disposal Warning



Warning: Ultimate disposal of this product should be handled according to all national laws and regulations.

No. 26 AWG Warning



Warning: To reduce the risk of fire, use only No. 26 AWG or larger telecommunication line cord.

LAN and FXS Ports Connectors Warning



Warning: Do not connect the LAN and the FXS ports connectors directly to the Public Switched Telephone Network (PSTN), to an off premise application, an out of plant application, any exposed plant application, or to any equipment other than the intended application, connection may result in a safety hazard, and/or defective operation and/or equipment damage.

Exposed plant means where any portion of the circuit is subject to accidental contact with electric lighting or power conductors operating at a voltage exceeding 300V between conductors or is subject to lightning strikes.

Socket Outlet Warning



Warning: The socket outlet, if used, shall be located near the equipment and shall be easily accessible by the user.

Safety Recommendations

To insure general safety follow these guidelines:

- ▶ Do not open or disassemble this product.
- ▶ Do not get this product wet or pour liquids into it.
- ▶ Do not perform any action that creates a potential hazard to people or makes the equipment unsafe.



Caution: When using this equipment, basic safety precautions should always be followed to reduce the risk of fire, electric shock and injury to persons, including the following:

- Do not use this product near water, for example, near a bath tub, wash bowl, kitchen sink or laundry tub, in a wet basement or near a swimming pool.
- Avoid using a telephone (other than a cordless type) during an electrical storm. There may be a remote risk of electric shock from lightning.
- Do not use the telephone to report a gas leak in the vicinity of the leak.

Standard Hardware Information

The specifications and information regarding this product are subject to change without notice. Every effort is made to ensure the accuracy of this document. Because of ongoing product improvements and revisions, Media5 cannot guarantee its accuracy, nor can be responsible for errors or omissions. Please contact your Media5 sales representative to obtain the latest version of the technical specifications.

Industry Standard Protocols

The Mediatrix 4100 has been designed to support all major industry standards used today, as well as those that will eventually be implemented at a later date. Because of this specific design characteristic, the Mediatrix 4100 can be integrated with existing telephone, fax and data equipment such as PCs and routers.

Table 230: Industry Standard Protocols

Parameter	Description
Vocoders	• G.711 (a-law, u-law) • G.723.1a • G.726 (40, 32, 24, 16 kbit/s) • G.729a • G.729ab
IP Telephony Protocols	• SIP - RFC 3261
Real-Time Transport Protocols	• RTP/RTCP - RFC 1889, RFC 1890, RFC 2833, RFC 3389 • Hook Flash Relay (RFC 2833)
Network Management Protocols	• SNMPv3 • DHCP - RFC 2131, RFC 2132 • TFTP - RFC 1350, RFC 2347, RFC 2348, RFC 2349 • Syslog - RFC 3164 • HTTP 1.0 - RFC 1945 • Basic and digest HTTP authentication - RFC 2617
QoS	• ToS • DiffServ • 802.1p • 802.1Q • STUN - RFC 3489

Hardware Features

Display

- ▶ Power LED
- ▶ LANactivity LED

- ▶ Activity/In-Use LED indication on FXS ports
- ▶ Ready LED

Interfaces

- ▶ 1 x RJ-21X TELCO 25 pairs connector, analog phone/fax (FXS) interface (Mediatrix 4124).
- ▶ 8 x RJ-11 connectors, analog phone/fax (FXS) interface (Mediatrix 4108).
- ▶ 16 x RJ-11 connectors, analog phone/fax (FXS) interface (Mediatrix 4116).
- ▶ 1 x RJ-11 connector, PSTN bypass.
- ▶ 2 x RJ-45 connectors, 10/100 BaseT Ethernet access (autosense: up to 100 Mbits).

Power

- ▶ AC: Standard power cord receptacle (IEC 320 – C14) for universal AC input internal SMPS.
- ▶ Seamless switch over period if the client UPS detects a power loss and activates within 8 ms.

Casing / Installation

- ▶ Casing: Desktop (Plastic ABS UL94 5V).
- ▶ Installation: rack-mountable, 1U size.

Product Architecture Details

- ▶ Supports up to 24 (depending on model) concurrent communications using any vocoders.
- ▶ DSP-based DTMF detection, generation and synthesis.
- ▶ DSP-based echo cancellation (G.168).
- ▶ DSP-based fax/data relay.
- ▶ Embedded operating system with 32-bit real-time multitasking Kernel.
- ▶ Embedded IPv4 TCP/IP stack with configurable QoS implemented by:
 - ToS byte at Network layer 3
 - 802.1p at Data Link layer 2
- ▶ Network parameters assigned via DHCP

Real Time Fax Router Technical Specifications

Automatic selection between voice and fax.

Table 231: Fax Technical Specifications

Parameter	Description
Protocols	Group 3 Fax Clear channel (G.711), G.726, or T.38 Real Time Fax Over IP protocol Stack
Fax Data Compression	MH
Fax Transmission	Up to 14.4 kbps

Analog Line Interface (FXS)

- ▶ RJ21X connector (Mediatix 4124)
- ▶ RJ-11 connectors (Mediatix 4108 and Mediatix 4116)
- ▶ Direct connection to a fax machine or telephone (Internal installation and internal cabling) (Mediatix 4108 and Mediatix 4116)
- ▶ DC feeding of the access line protected for over voltage
- ▶ Loop current detection and hook flash detection capable
- ▶ Generation of Selective Ring

Table 232: Analog Line Interface

Parameter	Description
Trunk Type	Loop Start: capable of Wink and Immediate signalization
Ring Source	45 VRMS max @ 20 up to 50 Hz (selectable) sine signal
Nominal Impedance	BellCore compliant 600/900 ohms default setting. Impedance Software Configurable.
Ring Drive Capacity	Up to 3 ringer equivalents (3 RENs) per port.
Loop Current Range	15 to 32 mA factory set. Default 20 mA regulated.
Ring Trip Detection Time	2 ring cycles max
On Hook Voltage	-48 VDC
Frequency Response	200 Hz to 3400 Hz ± 3 dB (Tx/Rx)
Return Loss	500-3200 Hz: 30 dB

Audio Specifications

- ▶ Software input and output level adjustable within the range of -30 dB to +20 dB.
- ▶ Software-adjustable dynamic and static jitter buffer protection.
- ▶ Programmable by country: Call progress tone generation including dial tone, busy tone, ringback and error tones.
- ▶ DSP-based echo control device.
- ▶ Silence detection/suppression level software adjustable.

DTMF Tone Detection

Table 233: DTMF Tone Detection

Parameter	Description
16-Digit DTMF Decoding	0 to 9, *, #, A, B, C, D
Permitted Amplitude Tilt	High frequency can be +4 dB to -8 dB relative to low frequency
Dynamic Range	-25 dBm to 0 dBm per tone
Frequency Accept	$\pm 1.5\%$ of nominal frequencies
Minimum Tone Duration	40 ms, can be increased with software configuration

Table 233: DTMF Tone Detection (Continued)

Parameter	Description
Interdigit Timing	Detects like digits with a 40 ms interdigit delay

DTMF Tone Generation

Table 234: DTMF Tone Generation

Parameter	Description
Per Frequency Nominal	-8 dBm to -5 dBm
Frequency Deviation	Within 1.5% of nominal values

MTBF Value

The Mean Time Before Failure (MTBF) value of the Mediatrix 4100 is 180,000 (Mediatrix 4124), 190,000 (Mediatrix 4116), and 200,000 (Mediatrix 4108) hours at 25 degrees Celsius ambient temperature. It has been defined using RelCalc v5.0, Bellcore method (LimitedStress - Method I, Case 3), Desktop unit.

Power Consumption

Measurements at the DC input

Table 235: Power Consumption at the DC Input

Parameter	Description
Idle Mode: 120Vac	0.54A 43W
Idle Mode: 240Vac	0.36A 43W
24 Extensions Off-Hook (worst case): 120Vac	0.58A 66W
24 Extensions Off-Hook (worst case): 240Vac	0.5A 66W

Operating Environment

Table 236: Operating Environment

Parameter	Description
Operating Temperature	0°C to 40°C
Humidity	Up to 85 %, non-condensing
Storage	-20°C to +70°C

Dimensions and Weight

Table 237: Dimensions and Weight

Parameter	Description
Height	4.4 cm (1.74 in.) approx.
Width	43 cm (17.19 in.) approx.
Depth	21 cm (8.4 in.) approx.
Weight	1.7 kg (3.7 lbs)

Warranty

All Media5 products carry Media5's standard three-year hardware and software warranty. An extended warranty is available.



Cabling Considerations

This Appendix describes the pin-to-pin connections for cables used with the Mediatix 4100.



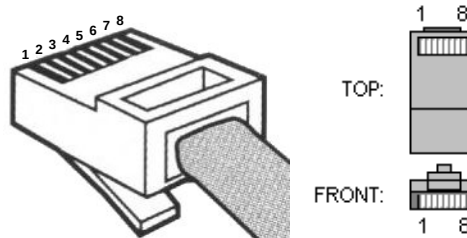
Warning: To reduce the risk of fire, use only No. 26 AWG or larger telecommunication line cord.

RJ-45 Cable

The RJ-45 connector is commonly used for network cabling and for telephony applications. It is used to wire both ends identically so the signals pass straight through.

RJ-45 cabling is also known as Twisted-pair Ethernet (TPE), Unshielded twisted pair (UTP) and 10/100 Base-T.

Figure 94: RJ-45 Cable



Straight Through Cable

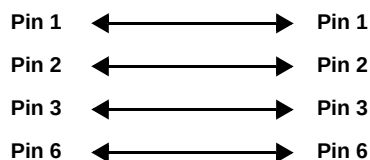
A RJ-45 straight through cable is used to connect a computer to a network device. For example straight through cables are the type of cables that connect a computer to a network hub, network switch, and network routers.

Table 238: RJ-45 Pinout Information

Pin #	Function	Colour Coding	
		EIA/TIA 568A	EIA/TIA 568B AT&T 258A
1	Transmit +	White with green stripe	White with orange stripe
2	Transmit -	Green with white stripe or solid green	Orange with white stripe or solid orange
3	Receive +	White with orange stripe	White with green stripe
4	N/A	Blue with white stripe or solid blue	Blue with white stripe or solid blue
5	N/A	White with blue stripe	White with blue stripe
6	Receive -	Orange with white stripe or solid orange	Green with white stripe or solid green
7	N/A	White with brown stripe or solid brown	White with brown stripe or solid brown
8	N/A	Brown with white stripe or solid brown	Brown with white stripe or solid brown

The RJ-45 cable uses two pairs of wires: one pair for transmission and the second pair for reception. It is wired so that pins 1 & 2 are on one twisted pair and pins 3 & 6 are on a second pair according to common wiring standards which meet the EIA/TIA T568A and T568B requirements.

Figure 95: Straight Through Connectivity



Pin Name And Function

The following is the meaning of each pin in a RJ-45 cable.

Table 239: Pin Name and Function

Pin #	Name	Function
1	Transmit Data Plus	The positive signal for the TD differential pair. This signal contains the serial output data stream transmitted onto the network.
2	Transmit Data Minus	The negative signal for the TD differential pair. This contains the same output as pin 1.
3	Receive Data Plus	The positive signal for the RD differential pair. This signal contains the serial input data stream received from the network.
4	not connected	
5	not connected	
6	Receive data minus	The negative signal for the RD differential pair. This signal contains the same input as pin 3.
7	not connected	
8	not connected	

Crossover Cable

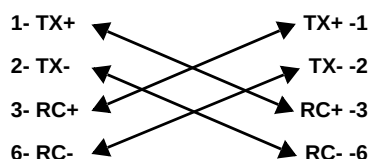
A RJ-45 crossover cable is used when only two systems are to be connected to each other, peer to peer, at the Ethernet Cards by “crossing over” (reversing) their respective pin contacts. An example would be connecting two computers together to create a network. The crossover eliminates the need for a hub when connecting two computers. A crossover cable may also be required when connecting a hub to a hub, or a transceiver to transceiver or repeater to repeater. When connecting a hub to a transceiver, a straight through cable is always used.



Note: This is not an IEEE supported configuration and should be used for test purposes only.

A crossover cable is sometimes called a null modem. The coloured wires at either end are put into different pin numbers, or crossed over.

Figure 96: Crossover Connectivity



RJ21X Connector (Mediatrix 4124)

The Mediatrix 4100 uses the RJ21X, or "Amphenol-type", connector as network interface with the Key Service Unit (KSU) or PBX Main Distribution Frame. The Mediatrix 4100 can thus be used in a variety of situations:

- ▶ It can be directly connected to a PBX via analog lines. These analog lines would be wired in a RJ21X cable.
- ▶ The PBX may have some analog lines connected to the SCN, while other lines could be connected to the IP network via the Mediatrix 4100.

See ["Placing a Call" on page 4](#) for a few examples of Mediatrix 4100 use.

Creating a RJ21X Cable

The Mediatrix 4100 can use from 1 to 24 single or multiple-pair circuits bridged to the network or other connected equipment. These circuits are usually wired in a RJ21X cable and then plugged into the RJ21X connector of the Mediatrix 4100.

To meet the Class B EMI requirements, a shielded cable and connector shall be used. The shield shall be an overall aluminium foil and be connected to the connector shield.



Note: Installation of the analog lines via a RJ21X cable shall be done by a qualified technician.

The cable may be directly connected to the PBX or cross-connected via a Bix termination and cross-connect system. In fact, depending on the technician involved, the scenarios may vary.

RJ21X Pinout

Pins 1 (ring) and 26 (tip) of the ribbon connector are considered position 1. Pins 2 (ring) and 27 (tip) are position 2. This pairing continues through 24 pairs.

Table 240: RJ21X Pinout

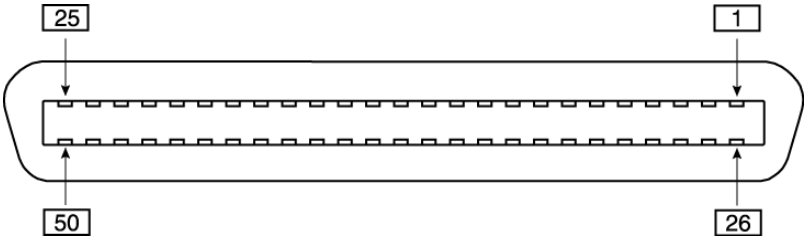
Line	Ring	Tip
1 (with Bypass)	pin1	pin26
2	pin2	pin27
3	pin3	pin28
4	pin4	pin29
5	pin5	pin30
6	pin6	pin31
7	pin7	pin32
8	pin8	pin33
9	pin9	pin34
10	pin10	pin35
11	pin11	pin36
12	pin12	pin37
13	pin13	pin38
14	pin14	pin39
15	pin15	pin40
16	pin16	pin41

Table 240: RJ21X Pinout (Continued)

Line	Ring	Tip
17	pin17	pin42
18	pin18	pin43
19	pin19	pin44
20	pin20	pin45
21	pin21	pin46
22	pin22	pin47
23	pin23	pin48
24	pin24	pin49

 **Note:** Pins 25 and 50 are unused.

Figure 97: Front View of the RJ21X Receptacle – Pinout



RJ-11 (Telephone) Cable

The RJ-11 cable is commonly used for telephone connection.



Caution: Do not plug a phone jack connector into an RJ-45 port.

Wiring Conventions

For telephone connections, a cable requires one pair of wires. Each wire is identified by different colors. For example, one wire might be red and the other, red with white stripes. Also, an RJ-11 connector must be attached to both ends of the cable.

Each wire pair must be attached to the RJ-11 connectors in a specific orientation. The following figure illustrates how the pins on the RJ-11 connector are numbered. Be sure to hold the connectors in the same orientation when attaching the wires to the pins.

Figure 98: RJ-11 Connector Pin Numbers

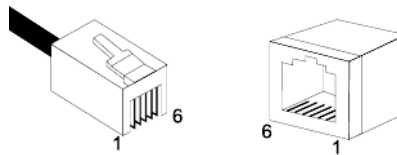


Table 241: RJ-11 Pinout Information

Pin #	Function
1	Not used
2	Not used
3	Ring
4	Tip
5	Not used
6	Not used

The RJ-11 pair of wires is wired so that pins 3 and 4 are connected to the Ring and Tip, which meets the following requirements:

- ▶ EIA/TIA-IS 968
- ▶ CS-03 Issue 8, Part III requirements.



Warning: The RJ-11 cable should comply with UL 1863 and CSA C22.2 No 233 standards.

Loop Current vs Cable Length

When installing the Mediatrix 4100, be sure that the cable length is not too long for the loop current. Considering the following operating parameters:

- ▶ Standard analog devices such as telephones or faxes with typically 300 Ohms of DC impedance.
- ▶ Cable with 85 Ohms/Km (AWG26)
- ▶ 0 dBm signal at interface

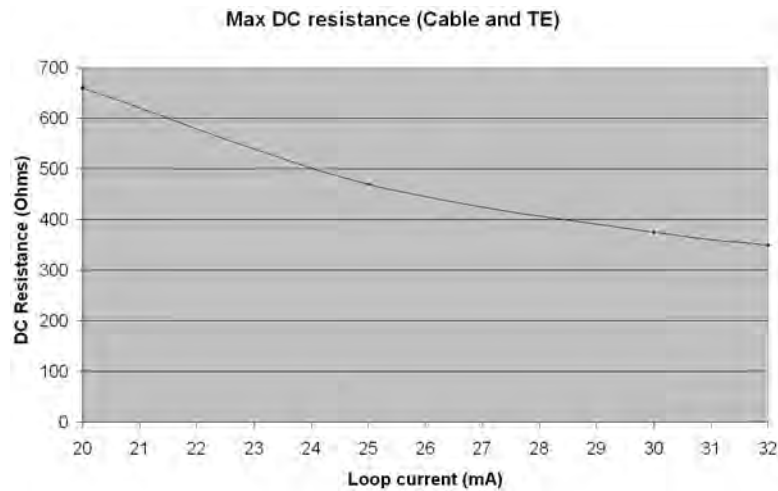
Observe the following values:

Table 242: Max DC Resistance

Loop Current in mA	Maximum Cable Length in Meters	Rdc Max in Ohms
20	2100	660
25	1000	470
30	450	375
32	300	350

Rdc max includes both equipment (standard analog telephones or faxes) and cable DC resistance. This is illustrated in [Figure 99 on page 384](#).

Figure 99: Max DC Resistance





Country-Specific Parameters

The following parameters differ depending on the country in which you are.

Definitions

The following are some useful definitions.

Table 243: Definitions

Term	Description
Dial Tone	Indicates the line is ready to receive dialing.
Busy Tone	Indicates the line or equipment is in use, engaged or occupied.
Ringback Tone	Indicates the called line is ringing out.
Special Information Tone	Identifies network-provided announcements.
Stutter Dial Tone	Notifies the user that they have a voice mail message when the phone does not or cannot have a message-waiting light.
Confirmation Tone	Confirms a command performed by the user (such as activate a service).
Receiver Off Hook (ROH) Tone	Indicates that the telephone is not hung up correctly.
Message Waiting Indicator Tone	Indicates there is a message waiting somewhere for the owner of the phone.
Network Congestion Tone	Indicates that all switching paths are busy, all toll trunks are busy, or there are equipment blockages.

Conventions

The following conventions apply to this Appendix.

Frequencies

- ▶ Symbol “*” means modulated. For instance: 425 Hz * 25 means 425 Hz modulated at 25 Hz.
- ▶ Symbol “+” means added. For instance: 425 Hz + 330 Hz means that both 425 Hz and 330 Hz sines are played at the same time.
- ▶ When a tone is composed of more than one frequency, if not otherwise specified, the given electrical level applies to each frequency taken separately.

Impedance

Impedance is the apparent resistance, in an electric circuit, to the flow of an alternating current, analogous to the actual electrical resistance to a direct current, being the ratio of electromotive force to the current.

When representing an impedance, the following applies:

- ▶ Symbol “//” means parallel.
- ▶ Symbol “+” means serial.

Furthermore, there are two types of impedances:

- ▶ Input Impedance
- ▶ Terminal Balance Return Loss (TBRL) Impedance

Input Impedance

Impedance of the Mediatrix 4100 at the Tip and Ring wires.

Terminal Balance Return Loss (TBRL) Impedance

Balance return loss attributable to transmission loss between two points. It is used to characterize an impedance balancing property of the 2-wire analog equipment port.

Each country has its own definition of the TBRL value. For instance, in North America, TIA/EIA 464 (and TIA/EIA 912) define two TBRL values:

- ▶ 600 Ω for “on-premise” or short loop ports.
- ▶ 350 Ω + (1000 Ω || 21 nF) for “off-premise” or long loop ports.

A wire length above 2.5 km is considered long loop according to TIA/EIA 912 section 6.4 (7)(b)).

In Europe, ETSI 300 439 also mentions a TBRL value. However, most European countries have different requirements regarding the TBRL Impedance. This is also true for other countries around the world. Each one of them has different requirements.

Line Attenuation

Values are given in dBr (decibel relative):

- ▶ A “+” for input means that the digital side is attenuated by x decibels relative to the analog side.
- ▶ A “+” for output means that the analog side is amplified by x decibels relative to the digital side.
- ▶ A “-” for input means that the digital side is amplified by x decibels relative to the analog side.
- ▶ A “-” for output means that the analog side is attenuated by x decibels relative to the digital side.

On-Off Sequences

Values in bold are “on” cycles, where tones are audible. Values in normal style are “off” cycles, where tones are not audible. When not otherwise specified, sequences repeat forever. A “x” symbol means that the sequences between parenthesis is repeated x times. The next cycle(s) repeat forever, unless otherwise specified. Values are in seconds.

For instance:

$3 * (\mathbf{0.1} - 0.1)$ then $\mathbf{0.6} - 1.0 - \mathbf{0.2} - 0.2$

means that the 0.1s on and 0.1s off sequence is repeated 3 times, afterwards the 0.6s on, 1.0s off, 0.2s on and 0.2s off sequence repeats forever.

Distinctive Ring

The distinctive ring service allows you to have three different numbers with each their own ring. The numbers ring through a single line coming into the business or residence and each number can be distinguished by the pattern of the ring. These ring patterns are made up of various combinations of ring bursts.

This feature uses the "*Alert-Info*" header from the initial INVITE of a call to know if the call requires a distinctive ringing.

The supported values of the "*Alert-Info*" are:

Table 244: Distinctive Ring Patterns

Alert-Info value	Ring Name	On – Off Sequence (s)
<http://127.0.0.1/Bellcore-dr2> or <Bellcore-dr2>	Bellcore-dr2	0.8 – 0.4, 0.8 – 4.0
<http://127.0.0.1/Bellcore-dr3> or <Bellcore-dr3>	Bellcore-dr3	0.4 – 0.2, 0.4 – 0.2, 0.8 – 4.0
<http://127.0.0.1/Bellcore-dr4> or <Bellcore-dr4>	Bellcore-dr4	0.3 – 0.2, 1.0 – 0.2, 0.3 – 4.0

The Mediatrix 4100 plays the default ring of the country selected if the *Alert-Info* value is not present or the value is not supported.



Note: Since the first pause of the distinctive ring is lower than 1 second, a splash ring followed by an Off of 1 second precedes the distinctive ring pattern.

Australia

The following parameters apply if you have selected Australia as location.

Australia 1

The following parameters apply if you have selected Australia1 as location.

Table 245: Australia 1 Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	425 Hz * 25	CONTINUOUS	-18 dBm
Busy Tone	425 Hz	0.375 – 0.375	-18 dBm
Ringback Tone	425 Hz * 25	0.4 – 0.2, 0.4 – 2.0	-17 dBm
Special Information Tone	950 Hz 1400 Hz 1800 Hz	0.333 0.333 0.333 – 1.0	-20 dBm
Stutter Dial Tone	425 Hz	CONTINUOUS	-18 dBm
Confirmation Tone	450 Hz	(0.15 – 0.15 – 0.15) x 2 End	-18 dBm
Receiver Off Hook (ROH) Tone	1400+2067+2467+2600 Hz	0.1 – 0.1	-21 dBm
Message Waiting Indicator Tone	425 Hz * 25	0.1 – 0.04, x72	-18 dBm
Network Congestion Tone	400 Hz	0.375 – 0.375	-18 dBm
Ring	AC: 45 VRMS, 25 Hz DC: 15 Vdc	0.4 – 0.2, 0.4 – 2.0	
Input Impedance	600 Ω		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			0 dBr
FXS Line Attenuation (Output)			-6 dBr

Australia 2

The following parameters apply if you have selected Australia2 as location.

Table 246: Australia 2 Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	425 Hz * 25	CONTINUOUS	-18 dBm
Busy Tone	425 Hz	0.375 – 0.375	-18 dBm
Ringback Tone	425 Hz * 25	0.4 – 0.2, 0.4 – 2.0	-17 dBm
Special Information Tone	950 Hz 1400 Hz 1800 Hz	0.333 0.333 0.333 – 1.0	-20 dBm
Stutter Dial Tone	425 Hz	CONTINUOUS	-18 dBm
Confirmation Tone	450 Hz	(0.15 – 0.15 – 0.15) x 2 End	-18 dBm
Receiver Off Hook (ROH) Tone	1400+2067+2467+2600 Hz	0.1 – 0.1	-21 dBm
Message Waiting Indicator Tone	425 Hz * 25	0.1 – 0.04, x72	-18 dBm
Network Congestion Tone	400 Hz	0.375 – 0.375	-18 dBm
Ring	AC: 45 VRMS, 25 Hz DC: 15 Vdc	0.4 – 0.2, 0.4 – 2.0	
Input Impedance	600 Ω		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			-3 dBr
FXS Line Attenuation (Output)			-6 dBr

Australia 3

The following parameters apply if you have selected Australia3 as location.

Table 247: Australia 3 Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	425 Hz * 25	CONTINUOUS	-18 dBm
Busy Tone	425 Hz	0.375 – 0.375	-18 dBm
Ringback Tone	425 Hz * 25	0.4 – 0.2, 0.4 – 2.0	-17 dBm
Special Information Tone	950 Hz 1400 Hz 1800 Hz	0.333 0.333 0.333 – 1.0	-20 dBm
Stutter Dial Tone	425 Hz	CONTINUOUS	-18 dBm
Confirmation Tone	450 Hz	(0.15 – 0.15 – 0.15) x 2 End	-18 dBm
Receiver Off Hook (ROH) Tone	1400+2067+2467+2600 Hz	0.1 – 0.1	-21 dBm
Message Waiting Indicator Tone	425 Hz * 25	0.1 – 0.04, x72	-18 dBm
Network Congestion Tone	400 Hz	0.375 – 0.375	-18 dBm
Ring	AC: 45 VRMS, 25 Hz DC: 15 Vdc	0.4 – 0.2, 0.4 – 2.0	
Input Impedance	220 Ω + 820 Ω // 115 nF		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			-3 dBr
FXS Line Attenuation (Output)			-9 dBr

Austria

The following parameters apply if you have selected Austria as location.

Austria 1

The following parameters apply if you have selected Austria1 as location.

Table 248: Austria Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	450 Hz	CONTINUOUS	-20 dBm
Busy Tone	450 Hz	0.3 – 0.3	-20 dBm
Ringback Tone	450 Hz	1.0 – 5.0	-20 dBm
Special Information Tone	950 Hz 1400 Hz 1800 Hz	0.33 0.33 0.33 – 1.0	-20 dBm
Stutter Dial Tone	450 Hz	(0.1 – 0.1) x 3, CONTINUOUS	-20 dBm
Confirmation Tone	450 Hz	(0.1 – 0.1) x 3 End	-20 dBm
Receiver Off Hook (ROH) Tone	1400+2060+2450+2600 Hz	0.1 – 0.1	-19 dBm
Message Waiting Indicator Tone	450 Hz	(0.1 – 0.1) x 10, CONTINUOUS	-20 dBm
Network Congestion Tone	450 Hz	(0.1 – 0.1) x10, CONTINUOUS	-20 dBm
Ring	AC: 45 VRMS, 50 Hz DC: 15 Vdc	1.0 – 5.0	
Input Impedance	270 Ω + 750 Ω // 150 nF		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			-3 dBr
FXS Line Attenuation (Output)			-10 dBr

Austria 2

The following parameters apply if you have selected Austria2 as location.

Table 249: Austria 2 Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	420 Hz	CONTINUOUS	-20 dBm
Busy Tone	420 Hz	0.4 – 0.4	-20 dBm
Ringback Tone	420 Hz	1.0 – 5.0	-20 dBm
Special Information Tone	950 Hz 1400 Hz 1800 Hz	0.33 0.33 0.33 – 1.0	-20 dBm
Stutter Dial Tone	380 + 420 Hz	CONTINUOUS	-20 dBm
Confirmation Tone	420 Hz	(0.1 – 0.1) x 3 End	-20 dBm
Receiver Off Hook (ROH) Tone	1400+2060+2450+2600 Hz	0.1 – 0.1	-19 dBm
Message Waiting Indicator Tone	420 Hz	(0.1 – 0.1) x 10 CONTINUOUS	-20 dBm
Network Congestion Tone	420 Hz	0.2 – 0.2	-20 dBm
Ring	AC: 45 VRMS, 50 Hz DC: 15 Vdc	1.0 – 5.0	
Input Impedance	270 Ω + 750 Ω // 150 nF		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			-3 dBr
FXS Line Attenuation (Output)			-10 dBr

Brazil

The following parameters apply if you have selected Brazil as location.

Table 250: Brazil Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	425 Hz	CONTINUOUS	-15 dBm
Busy Tone	425 Hz	0.25 – 0.25	-10 dBm
Ringback Tone	425 Hz	1.0 – 4.0	-15 dBm
Special Information Tone	950 Hz 1400 Hz 1800 Hz	(3 x 0.3 – 2 x 0.03) – 1.0	-15 dBm
Stutter Dial Tone	425 Hz	(0.1 – 0.1) x 3, CONTINUOUS	-15 dBm
Confirmation Tone	425 Hz	(0.1 – 0.1) x 3 End	-15 dBm
Receiver Off Hook (ROH) Tone	425 Hz	0.25 – 0.25	-10 dBm
Message Waiting Indicator Tone	425 Hz	(0.1 – 0.1) x 10, CONTINUOUS	-15 dBm
Reorder Tone	425 Hz	0.75 – 0.25, 0.25 – 0.25	-10 dBm
Network Congestion Tone	425 Hz	0.2 – 0.2	-10 dBm
Ring	AC: 45 VRMS, 25 Hz DC: 15 Vdc	1.0 – 4.0	
Input Impedance	900 Ω		
Default Caller ID	TELEBRAS_DTMF		
FXS Line Attenuation (Input)			0 dBr
FXS Line Attenuation (Output)			-7 dBr

Chile

The following parameters apply if you have selected Chile as location.

Chile 1

The following parameters apply if you have selected Chile1 as location.

Table 251: Chile 1 Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	400 Hz	CONTINUOUS	-10 dBm
Busy Tone	400 Hz	0.5 – 0.5	-10 dBm
Ringback Tone	400 Hz	1.0 – 3.0	-10 dBm
Special Information Tone	950 Hz 1400 Hz 1800 Hz	0.33 0.33 0.33 – 1.0	-10 dBm
Stutter Dial Tone	330 + 440 Hz	CONTINUOUS	-10 dBm
Confirmation Tone	425 Hz	(0.1 – 0.1) x 3 End	-10 dBm
Receiver Off Hook (ROH) Tone	425 Hz	0.25 – 0.25	-10 dBm
Message Waiting Indicator Tone	425 Hz	(0.1 – 0.1) x 10 CONTINUOUS	-10 dBm
Network Congestion Tone	400 Hz	0.2 – 0.2	-10 dBm
Ring	AC: 45 VRMS, 20 Hz DC: 15 Vdc	1.0 – 3.0	
Input Impedance	600 Ω		
Tbri-Impedance	600 Ω		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			0 dBr
FXS Line Attenuation (Output)			-7 dBr

Chile 2

The following parameters apply if you have selected Chile2 as location.

Table 252: Chile 2 Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	400 Hz	CONTINUOUS	-10 dBm
Busy Tone	400 Hz	0.5 – 0.5	-10 dBm
Ringback Tone	400 Hz	1.0 – 3.0	-10 dBm
Special Information Tone	950 Hz 1400 Hz 1800 Hz	0.33 0.33 0.33 – 1.0	-10 dBm
Stutter Dial Tone	330 + 440 Hz	CONTINUOUS	-10 dBm
Confirmation Tone	425 Hz	(0.1 – 0.1) x 3 End	-10 dBm
Receiver Off Hook (ROH) Tone	425 Hz	0.25 – 0.25	-10 dBm
Message Waiting Indicator Tone	425 Hz	(0.1 – 0.1) x 10 CONTINUOUS	-10 dBm
Network Congestion Tone	400 Hz	0.2 – 0.2	-10 dBm
Ring	AC: 45 VRMS, 20 Hz DC: 15 Vdc	1.0 – 3.0	
Input Impedance	900 Ω		
Tbri-Impedance	900 Ω		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			0 dBr
FXS Line Attenuation (Output)			-7 dBr

China

The following parameters apply if you have selected China as location.

Table 253: China Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	450 Hz	CONTINUOUS	-10 dBm
Busy Tone	450 Hz	0.35 – 0.35	-10 dBm
Ringback Tone	450 Hz	1.0 – 4.0	-10 dBm
Special Information Tone	950 Hz 1400 Hz 1800 Hz	0.33 0.33 0.33 – 1.0	-10 dBm
Stutter Dial Tone	450 Hz	0.4 – 0.04	-10 dBm
Confirmation Tone	450 Hz	(0.1 – 0.1) x 3, End	-10 dBm
Receiver Off Hook (ROH) Tone	950 Hz 950 Hz 950 Hz 950 Hz	15.0 – 15.0 – 15.0 15.0 – 15.0 – 15.0 15.0 – 15.0 – 15.0 15.0 – 15.0 – 15.0 – CONTINUOUS	-25 dBm -16 dBm -8 dBm -6 dBm
Message Waiting Indicator Tone	450 Hz	0.4 – 0.04	-10 dBm
Network Congestion Tone	450 Hz	0.7 – 0.7	-10 dBm
Ring	AC: 45 VRMS, 20 Hz DC: 15 Vdc	1.0 – 4.0	
Input Impedance	600 Ω		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			0 dBr
FXS Line Attenuation (Output)			-9 dBr

Czech Republic

The following parameters apply if you have selected Czech Republic as location.

Table 254: Czech Republic Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	425 Hz	0.33 – 0.33, 0.66 – 0.66	-12 dBm
Busy Tone	425 Hz	0.33 – 0.33	-12 dBm
Ringback Tone	425 Hz	1.0 – 4.0	-12 dBm
Special Information Tone	950 Hz 1400 Hz 1800 Hz	0.33 0.33 0.33 – 1.0	-12 dBm
Stutter Dial Tone	425 Hz	(0.165 – 0.165) x 3, 0.66 – 0.66	-12 dBm
Confirmation Tone	425 Hz	(0.1 – 0.1) x 3, End	-12 dBm
Receiver Off Hook (ROH) Tone	1400+2060+2450+2600 Hz	0.1 – 0.1	-19 dBm
Message Waiting Indicator Tone	425 Hz	(0.1 – 0.1) x 10, 0.33 – 0.33, 0.66 – 0.66	-12 dBm
Network Congestion Tone	425 Hz	0.165 – 0.165	-12 dBm
Ring	AC: 45 VRMS, 25 Hz DC: 15 Vdc	1.0 – 4.0	
Input Impedance	220 Ω + 820 Ω // 115 nF		
Default Caller ID	ETSI FSK		
FXS Line Attenuation (Input)			0 dBr
FXS Line Attenuation (Output)			-7 dBr

Denmark

The following parameters apply if you have selected Denmark as location.

Table 255: Denmark Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	425 Hz	CONTINUOUS	-15 dBm
Busy Tone	425 Hz	0.5 – 0.5	-10 dBm
Ringback Tone	425 Hz	1.0 – 4.0	-15 dBm
Special Information Tone	950 Hz 1400 Hz 1800 Hz	0.33 0.33 0.33 – 1.0	-15 dBm
Stutter Dial Tone	425 Hz	(0.1 – 0.1) x 3, CONTINUOUS	-15 dBm
Confirmation Tone	425 Hz	(0.1 – 0.1) x 3, End	-15 dBm
Receiver Off Hook (ROH) Tone	1400+2060+2450+2600 Hz	0.1 – 0.1	-19 dBm
Message Waiting Indicator Tone	425 Hz	(0.1 – 0.1) x 10, CONTINUOUS	-15 dBm
Network Congestion Tone	425 Hz	0.2 – 0.2	-10 dBm
Ring	AC: 45 VRMS, 25 Hz DC: 15 Vdc	1.0 – 4.0	
Input Impedance	300 Ω + 1000 Ω // 220 nF		
Default Caller ID	TDK_DTMF		
FXS Line Attenuation (Input)			0 dBr
FXS Line Attenuation (Output)			-6 dBr

France

The following parameters apply if you have selected France as location.

Table 256: France Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	440 Hz	CONTINUOUS	-16.9 dBm
Busy Tone	440 Hz	0.5 – 0.5	-19.9 dBm
Ringback Tone	440 Hz	1.5 – 3.5	-19.9 dBm
Special Information Tone	950 Hz 1400 Hz 1800 Hz	(3 x 0.3 – 2 x 0.03) – 1.0	-19.9 dBm
Stutter Dial Tone	440 Hz	(0.1 – 0.1) x 3, CONTINUOUS	-16.9 dBm
Confirmation Tone	440 Hz	(0.1 – 0.1) x 3, End	-16.9 dBm
Receiver Off Hook (ROH) Tone	1400+2060+2450+2600 Hz	0.1 – 0.1	-19 dBm
Message Waiting Indicator Tone	440 Hz	(0.1 – 0.1) x 10	-16.9 dBm
Network Congestion Tone	440 Hz	0.25 – 0.25	-19.9 dBm
Ring	AC: 45 VRMS, 50 Hz DC: 15 Vdc	1.5 – 3.5	
Input Impedance	215 Ω + 1000 Ω // 137 nF		
Default Caller ID	FRANCE: BELLCORE FRANCE_ETSI_FSK: ETSI_FSK FRANCE_ETSI_DTMF:ETSI_DTM F		
FXS Line Attenuation (Input)			+1.9 dBr
FXS Line Attenuation (Output)			-8.9 dBr

Germany

The following parameters apply if you have selected Germany as location.

Germany 1

The following parameters apply if you have selected Germany 1 as location.

Table 257: Germany 1 Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	425 Hz	CONTINUOUS	-16 dBm
Busy Tone	425 Hz	0.48 – 0.48	-16 dBm
Ringback Tone	425 Hz	1.0 – 4.0	-16 dBm
Special Information Tone	900 Hz 1400 Hz 1800 Hz	0.33 0.33 0.33 – 1.0	-16 dBm
Stutter Dial Tone	425 Hz	(0.1 – 0.1) x 3, CONTINUOUS	-16 dBm
Confirmation Tone	425 Hz	(0.1 – 0.1) x 3, End	-16 dBm
Receiver Off Hook (ROH) Tone	1400+2060+2450+2600 Hz	0.1 – 0.1	-19 dBm
Message Waiting Indicator Tone	425 Hz	(0.1 – 0.1) x 10	-16 dBm
Network Congestion Tone	425 Hz	0.24 – 0.24	-16 dBm
Ring	AC: 45 VRMS, 25 Hz DC: 15 Vdc	1.0 – 4.0	
Input Impedance	220 Ω + 820 Ω // 115 nF		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			-3 dBr
FXS Line Attenuation (Output)			-10 dBr

Germany 2

The following parameters apply if you have selected Germany 2 as location.

Table 258: Germany 2 Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	425 Hz	CONTINUOUS	-16 dBm
Busy Tone	425 Hz	0.48 – 0.48	-16 dBm
Ringback Tone	425 Hz	1.0 – 4.0	-16 dBm
Special Information Tone	900 Hz 1400 Hz 1800 Hz	0.33 0.33 0.33 – 1.0	-16 dBm
Stutter Dial Tone	425 Hz	(0.1 – 0.1) x 3, CONTINUOUS	-16 dBm
Confirmation Tone	425 Hz	(0.1 – 0.1) x 3, End	-16 dBm
Receiver Off Hook (ROH) Tone	1400+2060+2450+2600 Hz	0.1 – 0.1	-19 dBm
Message Waiting Indicator Tone	425 Hz	(0.1 – 0.1) x 10	-13 dBm
Network Congestion Tone	425 Hz	0.24 – 0.24	-13 dBm
Ring	AC: 45 VRMS, 25 Hz DC: 15 Vdc	1.0 – 4.0	
Input Impedance	220 Ω + 820 Ω // 115 nF		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			0 dBr
FXS Line Attenuation (Output)			-7 dBr

Germany 3

The following parameters apply if you have selected Germany 3 as location.

Table 259: Germany 3 Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	425 Hz	0.2 – 0.2, 0.2 – 0.2, 0.2 – 0.8	-16 dBm
Busy Tone	425 Hz	0.48 – 0.48	-16 dBm
Ringback Tone	425 Hz	1.0 – 4.0	-16 dBm
Special Information Tone	900 Hz 1400 Hz 1800 Hz	0.33 0.33 0.33 – 1.0	-16 dBm
Stutter Dial Tone	425 Hz	(0.1 – 0.1) x 3, CONTINUOUS	-16 dBm
Confirmation Tone	425 Hz	(0.1 – 0.1) x 3, End	-16 dBm
Receiver Off Hook (ROH) Tone	1400+2060+2450+2600 Hz	0.1 – 0.1	-19 dBm
Message Waiting Indicator Tone	425 Hz	(0.1 – 0.1) x 10	-16 dBm
Network Congestion Tone	425 Hz	0.24 – 0.24	-16 dBm
Ring	AC: 45 VRMS, 25 Hz DC: 15 Vdc	1.0 – 4.0	
Input Impedance	220 Ω + 820 Ω // 115 nF		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			-3 dBr
FXS Line Attenuation (Output)			-10 dBr

Hong Kong

The following parameters apply if you have selected Hong Kong as location.

Table 260: Hong Kong Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	350 + 440 Hz	CONTINUOUS	-13 dBm
Busy Tone	480 + 620 Hz	0.5 – 0.5	-13 dBm
Ringback Tone	440 + 480 Hz	0.4 – 0.2, 0.4 – 3.0	-13 dBm
Special Information Tone	950 Hz 1400 Hz 1800 Hz	0.33 0.33 0.33 – 1.0	-16 dBm
Stutter Dial Tone	350 + 440 Hz	(0.1 – 0.1) x 20, CONTINUOUS	-16 dBm
Confirmation Tone	350 + 440 Hz	0.1 – 0.1, 0.3 – End	-16 dBm
Receiver Off Hook (ROH) Tone	1400+2060+2450+2600 Hz	0.1 – 0.1	-19 dBm
Message Waiting Indicator Tone	350 + 440 Hz	(0.2 – 0.2, 0.5 – 0.2) x 4, CONTINUOUS	-16 dBm
Network Congestion Tone	480 + 620 Hz	0.25 – 0.25	-13 dBm
Ring	AC: 45 VRMS, 20 Hz DC: 15 Vdc	0.4 – 0.2, 0.4 – 3.0	
Input Impedance	600 Ω		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			0 dBr
FXS Line Attenuation (Output)			-6 dBr

Indonesia

The following parameters apply if you have selected Indonesia as location.

Table 261: Indonesia Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	425 Hz	CONTINUOUS	-9 dBm
Busy Tone	425 Hz	0.5 – 0.5	-9 dBm
Ringback Tone	425 Hz	1.0 – 4.0	-9 dBm
Special Information Tone	950 Hz 1400 Hz 1800 Hz	0.33 – 0.03, 0.33 – 0.03, 0.33 – 1.0	-9 dBm
Stutter Dial Tone	425 Hz	(0.1 – 0.1) x 3, CONTINUOUS	-9 dBm
Confirmation Tone	425 Hz	(0.1 – 0.1) x 3, End	-9 dBm
Receiver Off Hook (ROH) Tone	1400+2060+2450+2600 Hz	0.1 – 0.1	-19 dBm
Message Waiting Indicator Tone	950 Hz	0.33 – 0.03	-9 dBm
Network Congestion Tone	425 Hz	0.25 – 0.25	-9 dBm
Ring	AC: 45 VRMS, 25 Hz DC: 15 Vdc	1.0 – 4.0	
Input Impedance	600 Ω		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			-3 dBr
FXS Line Attenuation (Output)			-3 dBr

Israel

The following parameters apply if you have selected Israel as location.

Table 262: Israel Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	400 Hz	CONTINUOUS	-15 dBm
Busy Tone	400 Hz	0.5 – 0.5	-15 dBm
Ringback Tone	400 Hz	1.0 – 3.0	-15 dBm
Special Information Tone	1000 Hz 1400 Hz 1800 Hz	0.333 0.333 0.333 – 1.0	-15 dBm
Stutter Dial Tone	400 Hz	(0.1 – 0.1) x 3, CONTINUOUS	-15 dBm
Confirmation Tone	400 Hz	0.17 – 0.14, 0.34	-15 dBm
Receiver Off Hook (ROH) Tone	1440+2060+2452+2600 Hz	0.12 – 0.88	-20 dBm
Message Waiting Indicator Tone	400 Hz	(0.16 – 0.16) x 10, CONTINUOUS	-15 dBm
Network Congestion Tone	400 Hz	0.25 – 0.25	-15 dBm
Ring	AC: 45 VRMS, 25 Hz DC: 15 Vdc	1.0 – 3.0	
Input Impedance	600 Ω		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			0 dBr
FXS Line Attenuation (Output)			-9 dBr

Italy

The following parameters apply if you have selected Italy as location.

Table 263: Italy Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	425 Hz	0.2 – 0.2, 0.6 – 1.0	-13 dBm
Busy Tone	425 Hz	0.5 – 0.5	-13 dBm
Ringback Tone	425 Hz	1.0 – 4.0	-13 dBm
Special Information Tone	950 Hz 1400 Hz 1800 Hz	0.33 0.33 0.33 – 1.0	-20 dBm
Stutter Dial Tone	425 Hz	(0.1 – 0.1) x 3, 0.2 – 0.2, 0.6 – 1.0	-13 dBm
Confirmation Tone	425 Hz	(0.1 – 0.1) x 3, End	-13 dBm
Receiver Off Hook (ROH) Tone	1400+2060+2450+2600 Hz	0.1 – 0.1	-19 dBm
Message Waiting Indicator Tone	425 Hz	(0.1 – 0.1) x 10, 0.2 – 0.2, 0.6 – 1.0	-13 dBm
Network Congestion Tone	425 Hz	0.2 – 0.2	-13 dBm
Ring	AC: 45 VRMS, 25 Hz DC: 15 Vdc	1.0 – 4.0	
Input Impedance	180 Ω + 630 Ω // 60 nF		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			0 dBr
FXS Line Attenuation (Output)			-7 dBr

Japan

The following parameters apply if you have selected Japan as location.

Table 264: Japan Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	400 Hz	CONTINUOUS	-13 dBm
Busy Tone	400 Hz	0.5 – 0.5	-13 dBm
Ringback Tone	400 Hz * 16	1.0 – 2.0	-16 dBm
Special Information Tone	400 Hz	0.1 – 0.1	-13 dBm
Stutter Dial Tone	400 Hz	(0.1 – 0.1) x 3, CONTINUOUS	-13 dBm
Confirmation Tone	400 Hz	(0.1 – 0.1) x 3, End	-13 dBm
Receiver Off Hook (ROH) Tone	1400+2060+2450+2600 Hz	0.1 – 0.1	-19 dBm
Message Waiting Indicator Tone	400 Hz	(0.1 – 0.1) x 10, CONTINUOUS	-13 dBm
Network Congestion Tone	400 Hz	0.5 – 0.5	-13 dBm
Ring	AC: 45 VRMS, 20 Hz DC: 15 Vdc	1.0 – 2.0	
Input Impedance	600 Ω		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			0 dBr
FXS Line Attenuation (Output)			-9 dBr

Malaysia

The following parameters apply if you have selected Malaysia as location.

Table 265: Malaysia Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	425 Hz	CONTINUOUS	-14 dBm
Busy Tone	425 Hz	0.5 – 0.5	-18 dBm
Ringback Tone	425 Hz	0.4 – 0.2, 0.4 – 2.0	-16 dBm
Special Information Tone	900 Hz 1400 Hz 1800 Hz	1.0 1.0 1.0 – 1.0	-14 dBm
Stutter Dial Tone	425 Hz	(0.1 – 0.1) x 3, CONTINUOUS	-14 dBm
Confirmation Tone	425 Hz	(0.1 – 0.1) x 3 End	-14 dBm
Receiver Off Hook (ROH) Tone	1400+2060+2450+2600 Hz	0.1 – 0.1	-19 dBm
Message Waiting Indicator Tone	425 Hz	(0.1 – 0.1) x 10, CONTINUOUS	-14 dBm
Network Congestion Tone	425 Hz	0. – 0.25	-18 dBm
Ring	AC: 45 VRMS, 20 Hz DC: 15 Vdc	0.4 – 0.2, 0.4 – 2.0	
Input Impedance	600 Ω		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			0 dBr
FXS Line Attenuation (Output)			-9 dBr

Mexico

The following parameters apply if you have selected Mexico as location.

Table 266: Mexico Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	425 Hz	CONTINUOUS	-14 dBm
Busy Tone	425 Hz	0.25 – 0.25	-18 dBm
Ringback Tone	425 Hz	1.0 – 4.0	-16 dBm
Special Information Tone	900 Hz 1400 Hz 1800 Hz	1.0 1.0 1.0 – 1.0	-14 dBm
Stutter Dial Tone	425 Hz	(0.1 – 0.1) x 3, CONTINUOUS	-14 dBm
Confirmation Tone	425 Hz	(0.1 – 0.1) x 3, End	-14 dBm
Receiver Off Hook (ROH) Tone	1400+2060+2450+2600 Hz	0.1 – 0.1	-19 dBm
Message Waiting Indicator Tone	425 Hz	(0.1 – 0.1) x 10 CONTINUOUS	-14 dBm
Network Congestion Tone	425 Hz	0.25 – 0.25	-18 dBm
Ring	AC: 45 VRMS, 20 Hz DC: 15 Vdc	1.0 – 4.0	
Input Impedance	600 Ω		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			-3 dBr
FXS Line Attenuation (Output)			-3 dBr

Netherlands

The following parameters apply if you have selected Netherlands as location.

Table 267: Netherlands Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	425 Hz	CONTINUOUS	-17 dBm
Busy Tone	425 Hz	0.5 – 0.5	-17 dBm
Ringback Tone	425 Hz	1.0 – 4.0	-17 dBm
Special Information Tone	950 Hz 1400 Hz 1800 Hz	0.333 0.333 0.333 – 1.0	-17 dBm
Stutter Dial Tone	425 Hz	(0.1 – 0.1) x 3, CONTINUOUS	-17 dBm
Confirmation Tone	425 Hz	(0.1 – 0.1) x 3, End	-17 dBm
Receiver Off Hook (ROH) Tone	1400+2060+2450+2600 Hz	0.1 – 0.1	-19 dBm
Message Waiting Indicator Tone	425 Hz	(0.1 – 0.1) x 10 CONTINUOUS	-17 dBm
Network Congestion Tone	425 Hz	0.25 – 0.25	-17 dBm
Ring	AC: 45 VRMS, 20 Hz DC: 15 Vdc	1.0 – 4.0	
Input Impedance	270 Ω + 750 Ω // 150 nF		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			0 dBr
FXS Line Attenuation (Output)			-7 dBr

New Zealand

The following parameters apply if you have selected New Zealand as location.

Table 268: New Zealand Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	400 Hz	CONTINUOUS	-17 dBm
Busy Tone	400 Hz	0.5 – 0.5	-17 dBm
Ringback Tone	400 Hz + 450 Hz	0.4 – 0.2, 0.4 – 2.0	-19 dBm
Special Information Tone	1400 Hz	0.1 – 0.1	-17 dBm
Stutter Dial Tone	400 Hz	(0.1 – 0.1) x 3, CONTINUOUS	-17 dBm
Confirmation Tone	400 Hz	(0.1 – 0.1) x 3, End	-17 dBm
Receiver Off Hook (ROH) Tone	1400+2060+2450+2600 Hz	0.1 – 0.1	-19 dBm
Message Waiting Indicator Tone	400 Hz	(0.1 – 0.1) x12, CONTINUOUS	-17 dBm
Network Congestion Tone	400 Hz	0.25 – 0.25	-17 dBm
Ring	AC: 45 VRMS, 25 Hz DC: 15 Vdc	0.4 – 0.2, 0.4 – 2.0	
Input Impedance	300 Ω + 1000 Ω // 220 nF		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			-3 dBr
FXS Line Attenuation (Output)			-9 dBr

North America

The following parameters apply if you have selected North America as location.

North America 1

The following parameters apply if you have selected North America 1 as location.

Table 269: North America 1 Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	350+440 Hz	CONTINUOUS	-17 dBm
Busy Tone	480+620 Hz	0.5 – 0.5	-21 dBm
Ringback Tone	440+480 Hz	2.0 – 4.0	-19 dBm
Special Information Tone	950 Hz 1400 Hz 1800 Hz	0.33 0.33 0.33 – 1.0	-14 dBm
Stutter Dial Tone	350+440 Hz	(0.1 – 0.1) x 3, CONTINUOUS	-17 dBm
Confirmation Tone	350+440 Hz	(0.1 – 0.1) x 3, End	-17 dBm
Receiver Off Hook (ROH) Tone	1400+2060+2450+2600 Hz	0.1 – 0.1	-19 dBm
Message Waiting Indicator Tone	350+440 Hz	(0.1 – 0.1) x 10, CONTINUOUS	-17 dBm
Network Congestion Tone	480+620 Hz	0.25 – 0.25	-21 dBm
Ring	AC: 45 VRMS, 20 Hz DC: 15 Vdc	2.0 – 4.0	
Input Impedance	600 Ω		
Tbri-Impedance ^a	600 Ω		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			-3 dBr
FXS Line Attenuation (Output)			-3 dBr

a. TBRL-Impedance for "on-premise" or short loop ports.

North America 2

The following parameters apply if you have selected North America 2 as location.

Table 270: North America 2 Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	350+440 Hz	CONTINUOUS	-17 dBm
Busy Tone	480+620 Hz	0.5 – 0.5	-21 dBm
Ringback Tone	440+480 Hz	2.0 – 4.0	-19 dBm
Special Information Tone	950 Hz 1400 Hz 1800 Hz	0.33 0.33 0.33 – 1.0	-14 dBm
Stutter Dial Tone	350+440 Hz	(0.1 – 0.1) x 3, CONTINUOUS	-17 dBm
Confirmation Tone	350+440 Hz	(0.1 – 0.1) x 3, End	-17 dBm
Receiver Off Hook (ROH) Tone	1400+2060+2450+2600 Hz	0.1 – 0.1	-19 dBm
Message Waiting Indicator Tone	350+440 Hz	(0.1 – 0.1) x 10, CONTINUOUS	-17 dBm
Network Congestion Tone	480+620 Hz	0.25 – 0.25	-21 dBm
Ring	AC: 45 VRMS, 20 Hz DC: 15 Vdc	2.0 – 4.0	
Input Impedance	600 Ω		
Tbri-Impedance ^a	350 Ω + 1000 Ω // 210 nF		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			0 dBr
FXS Line Attenuation (Output)			0 dBr

a. TBRL-Impedance for “off-premise” or long loop ports (wire length longer than 2.5 km).

Russia

The following parameters apply if you have selected Russia as location.

Table 271: Russia Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	425 Hz	CONTINUOUS	-10 dBm
Busy Tone	425 Hz	0.4 – 0.4	-10 dBm
Ringback Tone	425 Hz	0.8 – 3.2	-10 dBm
Special Information Tone	950 Hz 1400 Hz 1800 Hz	0.333 0.333 0.333 – 1.0	-17 dBm
Stutter Dial Tone	425 Hz	(0.1 – 0.1) x 3, End	-10 dBm
Confirmation Tone	1400 Hz 2060 Hz 2450 Hz 2600 Hz	0.1 – 0.1	-19 dBm -19 dBm -19 dBm -19 dBm
Receiver Off Hook (ROH) Tone	425 Hz	3 x (0.1 – 0.1) , CONTINUOUS	-10 dBm
Message Waiting Indicator Tone	425 Hz	(0.1 – 0.1) x 10 CONTINUOUS	-10 dBm
Network Congestion Tone	425 Hz	0.2 – 0.2	-10 dBm
Ring	AC: 45 VRMS, 25 Hz DC: 15 Vdc	0.8 – 3.2	
Input Impedance	600 Ω		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			+2 dBr
FXS Line Attenuation (Output)			-2 dBr

South Africa

The following parameters apply if you have selected South Africa as location.

Table 272: South Africa Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	400 Hz	CONTINUOUS	-17 dBm
Busy Tone	400 Hz	0.5 – 0.5	-17 dBm
Ringback Tone	400 Hz + 450 Hz	0.4 – 0.2, 0.4 – 2.0	-19 dBm
Special Information Tone	1400 Hz	0.1 – 0.1	-17 dBm
Stutter Dial Tone	400 Hz	(0.1 – 0.1) x 3, CONTINUOUS	-17 dBm
Confirmation Tone	400 Hz	(0.1 – 0.1) x 3, End	-17 dBm
Receiver Off Hook (ROH) Tone	1400+2060+2450+2600 Hz	0.1 – 0.1	-19 dBm
Message Waiting Indicator Tone	400 Hz	(0.1 – 0.1) x12, CONTINUOUS	-17 dBm
Network Congestion Tone	400 Hz	0.25 – 0.25	-17 dBm
Ring	AC: 45 VRMS, 25 Hz DC: 15 Vdc	0.4 – 0.2, 0.4 – 2.0	
Input Impedance	300 Ω + 1000 Ω // 220 nF		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			-3 dBr
FXS Line Attenuation (Output)			-9 dBr

Spain

The following parameters apply if you have selected Spain as location.

Table 273: Spain Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	425 Hz	CONTINUOUS	-10 dBm
Busy Tone	425 Hz	0.2 – 0.2	-13 dBm
Ringback Tone	425 Hz	1.5 – 3.0	-13 dBm
Special Information Tone	950 Hz 1400 Hz 1800 Hz	0.33 0.33 0.33 – 1.0	-20 dBm
Stutter Dial Tone	425 Hz	(0.1 – 0.1) x 3, CONTINUOUS	-10 dBm
Confirmation Tone	425 Hz	(0.1 – 0.1) x 3, End	-10 dBm
Receiver Off Hook (ROH) Tone	1400+2060+2450+2600 Hz	0.1 – 0.1	-19 dBm
Message Waiting Indicator Tone	425 Hz	(0.1 – 0.1) x 10, CONTINUOUS	-10 dBm
Network Congestion Tone	425 Hz	0.2 – 0.2, 0.2 – 0.2, 0.2 – 0.6	-13 dBm
Ring	AC: 45 VRMS, 25 Hz DC: 15 Vdc	1.5 – 3.0	
Input Impedance	220 Ω + 820 Ω // 120 nF		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			0 dBr
FXS Line Attenuation (Output)			-7 dBr

Sweden

The following parameters apply if you have selected Sweden as location.

Table 274: Sweden Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	425 Hz	CONTINUOUS	-12.5 dBm
Busy Tone	425 Hz	0.25 – 0.25	-12.5 dBm
Ringback Tone	425 Hz	1.0 – 5.0	-12.5 dBm
Special Information Tone	950 Hz 1400 Hz 1800 Hz	0.33 0.33 0.33 – 1.0	-22 dBm
Stutter Dial Tone	425 Hz	(0.1 – 0.1) x 3, CONTINUOUS	-12.5 dBm
Confirmation Tone	425 Hz	(0.1 – 0.1) x 3, End	-12.5 dBm
Receiver Off Hook (ROH) Tone	1400+2060+2450+2600 Hz	0.1 – 0.1	-19 dBm
Message Waiting Indicator Tone	425 Hz	(0.1 – 0.1) x 10, CONTINUOUS	-12.5 dBm
Network Congestion Tone	425 Hz	0.25 – 0.75	-12.5 dBm
Ring	AC: 45 VRMS, 25 Hz DC: 15 Vdc	1.0 – 5.0	
Input Impedance	200 Ω + 1000 Ω // 100 nF		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			0 dBr
FXS Line Attenuation (Output)			-5 dBr

Switzerland

The following parameters apply if you have selected Switzerland as location.

Table 275: Switzerland Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	425 Hz	CONTINUOUS	-8 dBm
Busy Tone	425 Hz	0.5 – 0.5	-13 dBm
Ringback Tone	425 Hz	1.0 – 4.0	-13 dBm
Special Information Tone	950 Hz 1400 Hz 1800 Hz	0.333 0.333 0.333 – 1.0	-13 dBm
Stutter Dial Tone	425 Hz	(0.1 – 0.1) x 3, CONTINUOUS	-8 dBm
Confirmation Tone	425 Hz	(0.1 – 0.1) x 3, End	-8 dBm
Receiver Off Hook (ROH) Tone	1400+2060+2450+2600 Hz	0.1 – 0.1	-19 dBm
Message Waiting Indicator Tone	425 Hz	(0.1 – 0.1) x 10, CONTINUOUS	-8 dBm
Network Congestion Tone	425 Hz	0.2 – 0.2	-13 dBm
Ring	AC: 45 VRMS, 25 Hz DC: 15 Vdc	1.0 – 4.0	
Input Impedance	220 Ω + 820 Ω // 115 nF		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			0 dBr
FXS Line Attenuation (Output)			-6.5 dBr

Thailand

The following parameters apply if you have selected Thailand as location.

Table 276: Thailand Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	400 * 50 Hz	CONTINUOUS	-16 dBm
Busy Tone	400 Hz	0.5 – 0.5	-10 dBm
Ringback Tone	400 Hz	1.0 – 4.0	-10 dBm
Special Information Tone	950 Hz 1400 Hz 1800 Hz	0.33 0.33 0.33 – 1.0	-15 dBm
Stutter Dial Tone	400 * 50 Hz	(0.1 – 0.1) x 3, CONTINUOUS	-16 dBm
Confirmation Tone	400 * 50 Hz	(0.1 – 0.1) x 3, End	-16 dBm
Receiver Off Hook (ROH) Tone	1400+2060+2450+2600 Hz	0.1 – 0.1	-19 dBm
Message Waiting Indicator Tone	400 * 50 Hz	(0.1 – 0.1) x 10, CONTINUOUS	-16 dBm
Network Congestion Tone	400 Hz	0.3 – 0.3	-10 dBm
Ring	AC: 45 VRMS, 25 Hz DC: 15 Vdc	1.0 – 4.0	
Input Impedance	600 Ω		
Default Caller ID	BELLCORE		
FXS Line Attenuation (Input)			-3 dBr
FXS Line Attenuation (Output)			-3 dBr

United Arab Emirates

The following parameters apply if you have selected the United Arab Emirates 2 as location.

Table 277: United Arab Emirates 2 Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	350+440 Hz	CONTINUOUS	-13 dBm
Busy Tone	400 Hz	0.375 – 0.375	-13 dBm
Ringback Tone	425 Hz	0.4 – 0.2, 0.4 – 2.0	-13 dBm
Special Information Tone	950 Hz 1400 Hz 1800 Hz	0.33 0.33 0.33 – 1.0	-15 dBm -15 dBm -15 dBm
Stutter Dial Tone	350+440 Hz	(0.4 – 0.04) x 5, CONTINUOUS	-13 dBm
Confirmation Tone	400 Hz	(0.1 – 0.1) x 3 End	-13 dBm
Receiver Off Hook (ROH) Tone	1400+2060+2450+2600 Hz	(0.1 – 0.1)	-19 dBm
Ring	AC: 45 VRMS, 25 Hz DC: 15 Vdc	0.4 – 0.2, 0.4 – 2.0	
Input Impedance	600 Ω		
FXS Line Attenuation (Input)			3 dBr
FXS Line Attenuation (Output)			-3 dBr

UK

The following parameters apply if you have selected the United Kingdom as location.

Table 278: UK Parameters

Parameter	Value	On – Off Sequence (s)	Elect. Levels
Dial Tone	350+440 Hz	CONTINUOUS	-22 dBm
Busy Tone	400 Hz	0.375 – 0.375	-19 dBm
Ringback Tone	400+450 Hz	0.4 – 0.2, 0.4 – 2.0	-22 dBm
Special Information Tone	950 Hz 1400 Hz 1800 Hz	0.33 0.33 0.33 – 1.0	-19 dBm
Stutter Dial Tone	350+440 Hz	(0.1 – 0.1) x 3, CONTINUOUS	-22 dBm
Confirmation Tone	350+440 Hz	(0.1 – 0.1) x 3, End	-22 dBm
Receiver Off Hook (ROH) Tone	1400+2060+2450+2600 Hz	0.1 – 0.1	-19 dBm
Message Waiting Indicator Tone	350+440 Hz	(0.1 – 0.1) x 10, CONTINUOUS	-22 dBm
Network Congestion Tone	400 Hz	0.4 – 0.35, 0.225 – 0.525	-19 dBm
Ring	AC: 45 VRMS, 25 Hz DC: 15 Vdc	0.4 – 0.2, 0.4 – 2.0	
Input Impedance	300 Ω + 1000 Ω // 220 nF		
Default Caller ID	UK: BRITISH_TELECOM UK_BELLCORE: BELLCORE UK_CCA: CCA UK_ETSI_FSK: ETSI_FSK		
FXS Line Attenuation (Input)			-3 dBr
FXS Line Attenuation (Output)			-9 dBr

10 BaseT

An Ethernet local area network that works on twisted pair wiring.

100 BaseT

A newer version of Ethernet that operates at 10 times the speed of a 10 BaseT Ethernet.

Access Device

Device capable of sending or receiving data over a data communications channel.

A-Law

The ITU-T companding standard used in the conversion between analog and digital signals in PCM (Pulse Code Modulation) systems. A-law is used primarily in European telephone networks and contrasts with the North American mu (μ)-law standard. See also *mu (μ)-law*.

Analog Display Services Interface (ADSI)

Telecommunications protocol standard that enables alternate voice and data capability over the existing analog telephone network. This means that in addition to the familiar voice response audio interface (where you listen to voice recordings and make menu selections using the telephone keypad), you can now see the menu and information on the screen display and make selections using soft keys. To use ADSI, you would need an ADSI capable device (as you would if you want the caller ID service).

Area Code

The preliminary digits that a user must dial to be connected to a particular outgoing trunk group or line. In North America, an area code has three digits and is used with a NXX (office code) number. For instance, in the North American telephone number 561-955-1212, the numbers are defined as follows:

Table 279: North American Numbering Plan

No.	Description
561	Area Code, corresponding to a geographical zone in a non-LNP (Local Number Portability) network.
955	NXX (office code), which corresponds to a specific area such as a city region.
1212	Unique number to reach a specific destination.

Outside North America, the area code may have any number of digits, depending on the national telecommunication regulation of the country. In France, for instance, the numbering terminology is xZABPQ 12 34, where:

Table 280: France Numbering Plan

No.	Description
x	Operator forwarding the call. This prefix can be made of 4 digits.
Z	Geographical (regional) zone of the number (in France, there are five zones). It has two digits.
ABPQ	First four digits corresponding to a local zone defined by central offices.
12 34	Unique number to reach a specific destination.

In this context, the area code corresponds to the Z portion of the numbering plan. Because virtually every country has a different dialing plan nomenclature, it is recommended to identify the equivalent of an area code for the location of your communication unit.

Country Code (CC)

In international direct telephone dialing, a code that consists of 1-, 2-, or 3-digit numbers in which the first digit designates the region and succeeding digits, if any, designate the country.

Custom Local Area Signalling Services (CLASS)

One of an identified group of network-provided enhanced services. A CLASS group for a given network usually includes several enhanced service offerings, such as incoming-call identification, call trace, call blocking, automatic return of the most recent incoming call, call redial, and selective forwarding and programming to permit distinctive ringing for incoming calls.

Digital Signal Processor (DSP)

Specialized computer chip designed to perform speedy and complex operations on digitized waveforms. Useful in processing sound (like voice phone calls) and video.

Digital Subscriber Lines (DSL)

A technology for bringing high-bandwidth information to homes and small businesses over ordinary copper telephone lines. xDSL refers to different variations of DSL, such as ADSL, HDSL, and RADSL.

Domain Name Server (DNS)

Internet service that translates domain names into IP addresses. To use a domain name, a DNS service must translate the name into the corresponding IP address. For instance, the domain name *www.example.com* might translate to 198.105.232.4.

Dual-Tone Multi-Frequency (DTMF)

In telephone systems, multi-frequency signalling in which a standard set combinations of two specific voice band frequencies, one from a group of four low frequencies and the other from a group of four higher frequencies, are used. Although some military telephones have 16 keys, telephones using DTMF usually have 12 keys. Each key corresponds to a different pair of frequencies. Each pair of frequencies corresponds to one of the ten decimal digits, or to the symbol “#” or “*”, the “*” being reserved for special purposes.

Dynamic Host Configuration Protocol (DHCP)

TCP/IP protocol that enables PCs and workstations to get temporary or permanent IP addresses (out of a pool) from centrally-administered servers.

Echo Cancellation

Technique that allows for the isolation and filtering of unwanted signals caused by echoes from the main transmitted signal.

Far End Disconnect

Refers to methods for detecting that a remote party has hung up. This is also known as Hangup Supervision. There are several methods that may be used by a PBX/ACD/CO to signal that the remote party has hung up, including cleardown tone, or a wink.

Federal Communications Commission (FCC)

U.S. government regulatory body for radio, television, interstate telecommunications services, and international services originating in the United States.

Foreign Exchange Service/Station (FXS)

A network-provided service in which a telephone in a given local exchange area is connected, via a private line, to a central office in another, i.e., “foreign”, exchange, rather than the local exchange area's central office. This is the station (telephone) end of an FX circuit. An FXS port will provide dial tone and ring voltage.

G.711

ITU-T recommendation for an algorithm designed to transmit and receive A-law PCM (Pulse Code Modulation) voice at digital bit rates of 48 kbps, 56 kbps, and 64 kbps. It is used for digital telephone sets on digital PBX and ISDN channels.

G.723.1

A codec that provides the greatest compression, 5.3 kbps or 6.3 kbps; typically specified for multimedia applications such as H.323 videoconferencing.

G.726

Nn implementation of ITU-T G.726 standard for conversion linear or A-law or μ -law PCM to and from a 40, 32, 24 or 16 kbit/s channel.

G.729

A codec that provides near toll quality at a low delay which uses compression to 8 kbps (8:1 compression rate).

Gateway

A device linking two different types of networks that use different protocols (for example, between the packet network and the Public Switched Telephone Network).

Impedance

Impedance is the apparent resistance, in an electric circuit, to the flow of an alternating current, analogous to the actual electrical resistance to a direct current, being the ratio of electromotive force to the current.

International Telecommunication Union (ITU)

Organization based in Geneva, Switzerland, that is the most important telecom standards-setting body in the world.

Internet-Drafts

Internet-Drafts are working documents of the IETF, its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet Protocol (IP)

A standard describing software that keeps track of the Internet's addresses for different nodes, routes outgoing messages, and recognizes incoming messages.

Jitter

A distortion caused by the variation of a signal from its references which can cause data transmission errors, particularly at high speeds.

Layer 2

Layer 2 refers to the Data Link Layer of the commonly-referenced multilayered communication model, Open Systems Interconnection (OSI). The Data Link Layer is concerned with moving data across the physical links in the network.

The Data-Link Layer contains two sublayers that are described in the IEEE-802 LAN standards:

- ▶ Media Access Control (MAC)
- ▶ Logical Link Control (LLC)

Layer 3

Layer 3 refers to the Network layer of the commonly-referenced multilayered communication model, Open Systems Interconnection (OSI). The Network Layer is concerned with knowing the address of the neighbouring nodes in the network, selecting routes and quality of service, and recognizing and forwarding to the transport layer incoming messages for local host domains.

Light Emitting Diode (LED)

A semiconductor diode that emits light when a current is passed through it.

Local Area Network (LAN)

Data-only communications network confined to a limited geographic area, with moderate to high data rates. See also WAN.

Management Information Base (MIB)

Specifications containing definitions of management information so that networked systems can be remotely monitored, configured and controlled.

Management Server

Includes a web-based provisioning client, provisioning server, and SNMP proxy server used to manage all agents connected to the system. The Management Server provides Gateway provisioning, Monitoring, and Numbering Plan.

Media Access Control (MAC) Address

A layer 2 address, 6 bytes long, associated with a particular network device; used to identify devices in a network; also called hardware or physical address.

Mu (μ)-Law

The PCM (Pulse Code Modulation) voice coding and companding standard used in Japan and North America. See also *A-Law*.

Network

A group of computers, terminals, and other devices and the hardware and software that enable them to exchange data and share resources over short or long distances. A network can consist of any combination of local area networks (LAN) or wide area networks (WAN).

Off-hook

A line condition caused when a telephone handset is removed from its cradle.

On-hook

A line condition caused when a telephone handset is resting in its cradle.

Packet

Includes three principal elements: control information (such as destination, origin, length of packet), data to be transmitted, and error detection. The structure of a packet depends on the protocol.

Plain Old Telephone System (POTS)

Standard telephone service used by most residential locations; basic service supplying standard single line telephones, telephone lines, and access to the public switched network.

Port

Network access point, the identifier used to distinguish among multiple simultaneous connections to a host.

Portable Operating System Interface (POSIX)

POSIX is a set of standard operating system interfaces based on the UNIX operating system. The need for standardization arose because enterprises using computers wanted to be able to develop programs that could be moved among different manufacturer's computer systems without having to be recoded.

Private Branch Exchange (PBX)

A small to medium sized telephone system and switch that provides communications between onsite telephones and exterior communications networks.

Programmable Read-Only Memory (PROM)

A memory chip where data is written only once as it remains there forever. Unlike RAM, PROMs retain their contents when the computer is turned off.

Protocol

A formal set of rules developed by international standards bodies, LAN equipment vendors, or groups governing the format, control, and timing of network communications. A set of conventions dealing with transmissions between two systems. Typically defines how to implement a group of services in one or two layers of the OSI reference model. Protocols can describe low-level details of machine-to-machine interfaces or high-level exchanges between allocation programs.

Proxy Server

An intermediary program that acts as both a server and a client for the purpose of making requests on behalf of other clients. Requests are serviced internally or by passing them on, possibly after translation, to other servers. A proxy interprets, and, if necessary, rewrites a request message before forwarding it.

Public Switched Telephone Network (PSTN)

The local telephone company network that carries voice data over analog telephone lines.

Quality of Service (QoS)

Measure of the telephone service quality provided to a subscriber. This could be, for example, the longest time someone should wait after picking up the handset before they receive dial tone (three seconds in most U.S. states).

Real Time Control Protocol (RTCP)

RTCP is the control protocol designed to work in conjunction with RTP. It is standardized in RFC 1889 and 1890. In an RTP session, participants periodically send RTCP packets to convey feedback on quality of data delivery and information of membership.

Realtime Transport Protocol (RTP)

An IETF standard for streaming realtime multimedia over IP in packets. Supports transport of real-time data like interactive voice and video over packet switched networks.

Registrar Server

A server that accepts REGISTER requests. A registrar is typically co-located with a proxy or redirect server and MAY offer location services.

Request for Comment (RFC)

A Request for Comments (RFC) is a formal document from the IETF that is the result of committee drafting and subsequent review by interested parties. Some RFCs are informational in nature. Of those that are intended to become Internet standards, the final version of the RFC becomes the standard and no further comments or changes are permitted. Change can occur, however, through subsequent RFCs that supersede or elaborate on all or parts of previous RFCs.

Ring

One of the two wires (the two are Tip and Ring) needed to set up a telephone connection. See *Tip*.

Router

A specialized switching device which allows customers to link different geographically dispersed local area networks and computer systems. This is achieved even though it encompasses different types of traffic under different protocols, creating a single, more efficient, enterprise-wide network.

Switched Circuit Network (SCN)

A communication network, such as the public switched telephone network (PSTN), in which any user may be connected to any other user through the use of message, circuit, or packet switching and control devices.

Server

A computer or device on a network that works in conjunction with a client to perform some operation.

Session Description Protocol (SDP)

Describes multimedia sessions for the purpose of session announcement, session invitation and other forms of multimedia session initiation. SDP communicates the existence of a session and conveys sufficient information to enable participation in the session. SDP is described in RFC 2327.

Session Initiation Protocol (SIP)

A protocol for transporting call setup, routing, authentication, and other feature messages to endpoints within the IP domain, whether those messages originate from outside the IP cloud over SCN resources or within the cloud.

Simple Network Management Protocol (SNMP)

A standard of network management that uses a common software agent to manage local and wide area network equipment from different vendors; part of the Transmission Control Protocol / Internet Protocol (TCP/IP) suite and defined in RFC 1157.

Simple Network Time Protocol (SNTP)

SNTP, which is an adaptation of the Network Time Protocol (NTP), is widely used to synchronize computer clocks in the global Internet. It provides comprehensive mechanisms to access national time and frequency dissemination services, organize the time-synchronization subnet and adjust the local clock in each participating subnet peer. In most places of the Internet of today, NTP provides accuracies of 1-50 ms, depending on the characteristics of the synchronization source and network paths.

Stack

A set of network protocol layers that work together. The OSI Reference Model that defines seven protocol layers is often called a stack, as is the set of TCP/IP protocols that define communication over the Internet.

Subnet

An efficient means of splitting packets into two fields to separate packets for local destinations from packets for remote destinations in TCP/IP networks.

T.38

An ITU-T Recommendation for Real-time fax over IP. T.38 addresses IP fax transmissions for IP-enabled fax devices and fax gateways, defining the translation of T.30 fax signals and Internet Fax Protocols (IFP) packets.

Telephony

The science of translating sound into electrical signals, transmitting them, and then converting them back into sound.

Tip

The first wire in a pair of telephones wire. The second wire is called the "ring" wire. The tip is the conductor in a telephone pair cable which is usually connected to the positive side of a battery at the telephone company's central office. It is the telephone industry's equivalent of Ground in a normal electrical circuit. See also *Ring*.

Transmission Control Protocol/Internet Protocol (TCP/IP)

The basic communication language or protocol of the Internet. It can also be used as a communications protocol in a private network (either an intranet or an extranet).

Trivial File Transfer Protocol (TFTP)

A simplified version of FTP that transfers files but does not provide password protection, directory capability, or allow transmission of multiple files with one command.

User Datagram Protocol (UDP)

An efficient but unreliable, connectionless protocol that is layered over IP, as is TCP. Application programs are needed to supplement the protocol to provide error processing and retransmission of data. UDP is an OSI layer 4 protocol.

Voice Over IP (VoIP)

The technology used to transmit voice conversations over a data network using the Internet Protocol. Such data network may be the Internet or a corporate Intranet.

Wide Area Network (WAN)

A large (geographically dispersed) network, usually constructed with serial lines, that covers a large geographic area. A WAN connects LANs using transmission lines provided by a common carrier.

List of Acronyms

ABNF	Augmented Backus-Naur Form
ADSI	Analog Display Services Interface
AWG	American Wire Gauge
CE	Cummunauté européenne (French)
CNG	Comfort Noise Generator
CS-ACELP	Conjugate Structure-Algebraic Code Excited Linear Prediction
dB	Decibel
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name Server
DS	Differentiated Services
DSCP	Differentiated Services Code Point
DTMF	Dual Tone Multi-Frequency
FCC	Federal Communications Commission (USA)
FSK	Frequency Shift Keying
GMT	Greenwich Mean Time
HTTP	Hyper Text Transfer Protocol
Hz	Hertz
IEEE	Institute of Electrical & Electronics Engineers
IETF	Internet Engineering Task Force
IP	Internet Protocol
ITSP	Internet Telephony Service Provider
KSU	Key Service Unit
LAN	Local Area Network
LCD	Liquid Crystal Display
LED	Light Emitting Diode
MAC	Media Access Control
Mb/s	Megabits Per Second
MIB	Management Information Base
MTU	Maximum Transmission Unit
MWI	Message Waiting Indicator
NAT	Name Address Translation
OSI	Open Systems Interconnection
PBX	Private Branch eXchange
PCM	Pulse Code Modulation
PIN	Personal Identification Number
PSTN	Public Switched Telephone Network
QoS	Quality of Service
REN	Ringer Equivalence Number

RFC	Request For Comment
RTCP	Real Time Control Protocol
SCN	Switched Circuit Network
SDP	Session Description Protocol
SIP	Session Initiation Protocol
SMI	Structure of Management Information
SMPS	Switching Mode Power Supply
SSL	Secure Sockets Layer
STP	Spanning Tree Protocol
STUN	Simple Traversal of User Datagram Protocol (UDP) through Network Address Transla-
tion (NAT)	
TBRL	Terminal Balance Return Loss
TCP/IP	Transmission Control Protocol/Internet Protocol
TLS	Transport Layer Security
TPE	Twisted-Pair Ethernet
UDP	User Datagram Protocol
UTC	Universal Time Coordinated
UTP	Unshielded Twisted pair
VAC	Volts Alternating Current
VAD	Voice Activity Detection
VLAN	Virtual Local Area Network
VoIP	Voice over Internet Protocol
WAN	Wide Area Network
XML	eXtensible Markup Language



List of MIB Parameters

A

analogScnGwDialEnable	327
analogScnGwDtmfDuration	177, 240
analogScnGwInterDigitDialDelay	177, 240

C

certificateExpirationDate	189, 209
certificateName	188, 208
certificateSubjectCommonName	189, 208
checkTcpIpStackForSuccessfulBoot	22
configFileAutoUpdateOnRestartEnable	196
configFileAutoUpdatePeriod	197
configFileAutoUpdatePeriodicEnable	198
configFileAutoUpdateTimeRange	198
configFileAutoUpdateTimeUnit	197
configFileFetchingConfigSource	21
configFileFetchingDhcpSiteSpecificCode	150, 190
configFileFetchingFileLocation	191
configFileFetchingFileName	191
configFileFetchingHost	150, 189
configFileFetchingPort	150, 189
configFileFetchingSelectConfigSource	150, 189
configFileFetchingSpecificFileName	191
configFileFetchingStaticHost	150, 190
configFileFetchingStaticPort	150, 190
configFilePrivacyEnable	193
configFilePrivacyGenericSecret	193
configFilePrivacySpecificSecret	193
configFileTransferPassword	195, 196, 197
configFileTransferProtocol	194, 195, 196, 197
configFileTransferUsername	195, 196, 197
countryCustomizationToneOverride	181
countryCustomizationToneTone	181

D

dataIfAnalogCedDetectionBehavior	250
dataIfCedFaxToneEnable	250
dataIfClearChannelCodecPreferred	251
dataIfCngToneDetectionEnable	249
dataIfCodecT38Enable	254
dataIfCodecT38ProtectionLevel	255
dataIfT38FinalFramesRedundancy	255
dataIfT38NoSignalEnable	255
dataIfT38NoSignalTimeout	255
digitMapAllowedDigitMap	296
digitMapAllowedEnable	297
digitMapAllowedLineToApply	297
digitMapPrefixedDigitRemovalCount	296
digitMapPrependedString	297
digitMapProcessDigitsWhenPressed	296
digitMapRefusedDigitMap	297

digitMapRefusedEnable	297
digitMapRefusedLineToApply	297
digitMapSuffixStringToRemove	296
digitMapTimeoutCompletion	298
digitMapTimeoutFirstDigit	298
digitMapTimeoutInterDigit	298

E

emergencyCallUrgentGatewayDigitMap	304
emergencyCallUrgentGatewayEnable	304
emergencyCallUrgentGatewayTargetAddress	304

F

fxoWaitForCalleeToAnswerEnable	327
fxsBlankAnonymousCallerId	228
fxsByPassEnable	257
fxsCalleeHangupDelay	227
fxsCalleeHangupSupervision	227
fxsCallingNumberCriteria	229
fxsCallingNumberTransformation	229
fxsFlashHookDetectionDelayMax	225
fxsFlashHookDetectionDelayMin	225
fxsLoopCurrent	226
fxsLoopCurrentDropEnable	227
fxsPolarityAndDenialBehavior	228
fxsPowerDropOnDisconnectDuration	228

G

groupAdminState	195, 215, 216
groupReset	22
groupSetAdmin	22, 194, 195, 215, 216

H

httpServerAccess	25
httpServerDefaultPassword	32
httpServerEnable	25
httpServerPort	25
httpServerResetToDefaultPwd	32, 33
httpServerUsername	32

I

ifAdminInitialAdminState	224
ifAdminSetAdmin	223
imageAutoUpdateEnable	217, 219
imageAutoUpdateOnRestartEnable	217
imageAutoUpdatePeriod	218
imageAutoUpdateTimeRange	219
imageAutoUpdateTimeUnit	218
imageConfigSource	20
imageDhcpPrimarySiteSpecificCode	148, 209
imageDhcpSecondarySiteSpecificCode	148, 210
imageLocation	211, 212
imageLocationProvisionSource	211
imagePrimaryHost	20, 148, 209
imagePrimaryPort	20, 148, 209
imageSecondaryHost	20, 148, 209
imageSecondaryPort	20, 148, 209

imageSelectConfigSource	148, 209
imageSelectionFileLocation	211, 212
imageStaticPrimaryHost	210
imageStaticPrimaryPort	210
imageStaticSecondaryHost.....	210
imageStaticSecondaryPort	210
imageTransferPassword	216, 217, 218
imageTransferProtocol.....	215, 216, 217, 218
imageTransferUsername	216, 217, 218

L

lanStaticAddress	153
lanStaticNetworkMask	153
lineSelectionDigitMap	225
lineSelectionEnable	225
localHostAddress	20, 145, 146
localHostConfigSource	20
localHostDefaultRouter	20, 145
localHostDhcpServer	145
localHostDnsOverrideEnable	148
localHostPrimaryDns	20, 145
localHostSecondaryDns.....	20, 145
localHostSelectConfigSource	145
LocalHostSnmppPort	147
localHostSnmppPort	20, 147
localHostStaticSnmppPort	147
localHostStaticWanAddress.....	146, 265
localHostSubnetMask	20, 145
LocalHostWanAddress	146
localHostWanAddressSelectConfigSource	146, 265

M

msConfigSource	21
msDhcpSiteSpecificCode	149, 336
msEnable	335
msHost.....	20, 149, 335
msSelectConfigSource	149, 335
msStaticHost	149, 336
msStaticPort.....	149
msStaticTrapPort	149, 336
msTrapConfigInformation	201
msTrapPort	20, 149, 335
msTrapRetransmissionPeriod.....	335
msTrapRetransmissionRetryCount	335
mwiConfigActivation.....	332
mwiConfigFetchAddress	332
mwiConfigUserSubscriptionAddress.....	331
mwiConfigVoltageEnable	332
mwiExpirationTime.....	332
mwiFetchDigitMap	332
mwiSubscriptionCmdRefresh.....	332
mxDebugPcmCaptureEnable	353
mxDebugPcmCaptureEndpointNumber	353
mxDebugPcmCaptureIpAddress	353
mxInteropHttpUAHeaderConfig	195, 217

P

pinDialingDelay	326
-----------------------	-----

pinDialingEnable	326
pinDialingPin	326

Q

qosInteropUseVoiceQoSForRtcpEnable	339
qosSignalingDiffServ	337
qosSignalingIeee8021qEnable	338
qosSignalingIeee8021qUserPriority	338
qosT38FaxDiffServ	338
qosT38FaxIeee8021qEnable	338
qosT38FaxIeee8021qUserPriority	338
qosVlanIeee8021qDefaultUserPriority	340
qosVlanIeee8021qTaggingEnable	340
qosVlanIeee8021qVirtualLanID	340
qosVoiceDiffServ	337
qosVoiceIeee8021qEnable	338
qosVoiceIeee8021qUserPriority	338

S

sipAllowAudioAndImageNegotiationEnable	286
sipAllowMediaReactivationInAnswerEnable	285
sipCodecOrderInAnswer	286
sipConferenceServerURI	320
sipConferenceType	320
sipDebugContextSnapshotTime	354
sipDomain	259
sipEnforceOfferAnswerModel	284
sipHomeDomainProxyDhcpSiteSpecificCode	151, 167
sipHomeDomainProxyHost	151, 167
sipHomeDomainProxyPort	151, 167
sipHomeDomainProxyStaticHost	168
sipHomeDomainProxyStaticPort	168, 174
sipInteropAckUnsupportedInfoRequests	279
sipInteropAllowAsymmetricDtmfPayloadType	283
sipInteropAllowMultipleActiveMediaInAnswer	276
sipInteropAuthenticationQop	263
sipInteropAutomaticRejectionCode	329
sipInteropBehaviorOnT38InviteRejectedWith606	256
sipInteropBranchMatchingMethod	278
sipInteropCallWaitingToneControlViaSipInfo	283
sipInteropConferenceServerMechanism	320
sipInteropDefaultPublicationExpiration	270
sipInteropDefaultRegistrationExpiration	269
sipInteropDtmfTransportDuration	240
sipInteropDtmfTransportMethod	239
sipInteropEscapePoundInSipUriUsername	284
sipInteropIgnoreUsernameParam	283
sipInteropIgnoreViaBranchIdInCancelEnable	278
sipInteropInternationalCodeMappingEnable	280
sipInteropInternationalCodeMappingString	280
sipInteropLocalRingOnProvisionalResponse	277
sipInteropLockDnsSrvRecordPerCallEnable	175
sipInteropMaxForwardsValue	273
sipInteropMwiMessageSummaryValidation	332
sipInteropOnHoldAnswerSdpStreamDirection	276
sipInteropOnHoldSdpStreamDirection	275
sipInteropProxyAuthenticationUriParametersEnable	279
sipInteropReferredByConfig	274
sipInteropRegisterHomeDomainHostOverride	260

sipInteropRemoveOutboundProxyRouteHeader	280
sipInteropReplacesConfig	272
sipInteropReplacesVersion	273
sipInteropRetryFailedRegistration	281
sipInteropReuseCredentialEnable	175, 277
sipInteropRingingResponseCode	279
sipInteropSdpDirectionAttributeEnable	274
sipInteropSendUAHeaderEnable	260
sipInteropSessionTimersVersion	261
sipInteropSipOptionsMethodSupport	284
sipInteropSymmetricUdpSourcePortEnable	266
sipInteropT38NoSignalBehavior	255
sipInteropTransferVersion	271
sipInteropTransmissionTimeout	175, 273
sipInteropUAHeaderConfig	260
sipInteropUseDtmfPayloadTypeFoundInAnswer	282
sipInteropUseIuT38Format	280
sipInteropUsePAssertedHeader	282
sipInteropUseSipDomainInRequestURI	281
sipOutboundProxyConfig	170
sipOutboundProxyDhcpSiteSpecificCode	151, 169
sipOutboundProxyHost	151, 169
sipOutboundProxyPort	151, 169
sipOutboundProxyStaticHost	170
sipOutboundProxyStaticPort	170, 174
sipPenaltyBoxEnable	268
sipPenaltyBoxTime	175, 267
sipPort	259
sipPresenceCompositorDhcpSiteSpecificCode	172
sipPresenceCompositorHost	152, 171
sipPresenceCompositorPort	152, 171
sipPresenceCompositorStaticHost	172
sipPresenceCompositorStaticPort	172
sipPublicationCmdRefresh	270
sipPublicationProposedExpirationValue	270
sipPublicationRefreshTime	270
sipRegistrarDhcpSiteSpecificCode	151, 166
sipRegistrarHost	151, 165
sipRegistrarPort	151, 165
sipRegistrarStaticHost	166
sipRegistrarStaticPort	166, 174
sipRegistrationCmdRefresh	268
sipRegistrationProposedExpirationValue	269
sipReRegistrationTime	268
sipRtpUdpChecksumEnable	286
sipServerSelectConfigSource	151, 165, 167, 169, 171
sipTransportContactEnable	266
sipTransportEnable	265
sipTransportQValue	265
sipTransportRegistrationEnable	265
sipTrustedSourcesEnable	264
sipTrustedSourcesIpAddress	264
sipUAAuthPassword	263
sipUAAuthRealm	263
sipUAAuthUsername	263
sipUAAuthValidateRealm	263
sipUADisplayName	260
sipUAMainUsername	259
sipUAMaximumSessionExpirationDelay	261
sipUAMinimumSessionExpirationDelay	261

sipUAOtherAcceptedUsernames	260
sipUnitAuthPassword	263
sipUnitAuthRealm	263
sipUnitAuthUsername	263
sipUnitAuthValidateRealm	263
sipUnregisteredPortBehavior	224, 354
snmpAgentAccess	140
snmpAgentEnable	140
sntpConfigSource	21
sntpEnable	289
sntpHost	152, 290
sntpPort	152, 290
sntpSelectConfigSource	152, 290
sntpStaticHost	290
sntpStaticPort	290
sntpSynchronizationPeriod	289
sntpSynchronizationPeriodOnError	289
sntpTimeZoneString	198, 219, 291
stunEnable	288
stunKeepAliveInterval	288
stunQueryCacheDuration	288
stunQueryTimeout	288
stunStaticHost	288
stunStaticPort	288
subscriberServicesAttendedTransferEnable	318
subscriberServicesAttendedTransferStatus	318
subscriberServicesBlindTransferEnable	317
subscriberServicesBlindTransferStatus	317
subscriberServicesCallForwardOnBusyActivation	310
subscriberServicesCallForwardOnBusyDisableDigitMap	310
subscriberServicesCallForwardOnBusyEnable	310
subscriberServicesCallForwardOnBusyEnableDigitMap	310
subscriberServicesCallForwardOnBusyForwardingAddress	310
subscriberServicesCallForwardOnNoAnswerActivation	312
subscriberServicesCallForwardOnNoAnswerDisableDigitMap	312
subscriberServicesCallForwardOnNoAnswerEnable	312
subscriberServicesCallForwardOnNoAnswerEnableDigitMap	312
subscriberServicesCallForwardOnNoAnswerForwardingAddress	312
subscriberServicesCallForwardOnNoAnswerTimeout	312
subscriberServicesCallForwardUnconditionalActivation	308
subscriberServicesCallForwardUnconditionalDisableDigitMap	308
subscriberServicesCallForwardUnconditionalEnable	309
subscriberServicesCallForwardUnconditionalEnableDigitMap	308
subscriberServicesCallForwardUnconditionalForwardingAddress	308
subscriberServicesCallWaitingCancelDigitMap	314
subscriberServicesCallWaitingEnable	314
subscriberServicesCallWaitingPermanentDigitMapDisable	315
subscriberServicesCallWaitingPermanentDigitMapEnable	314
subscriberServicesCallWaitingStatus	314
subscriberServicesConferenceEnable	320
subscriberServicesConferenceStatus	320
subscriberServicesHoldEnable	307
subscriberServicesHoldStatus	307
subscriberServicesProcessingTrigger	305
subscriberServicesSecondCallEnable	307
subscriberServicesSecondCallStatus	307
sysAdminCommand	145, 215, 216
sysAdminDefaultSettingsEnable	22
sysAdminDownloadConfigFileStatus	192, 201
sysConfigBootpFlags	154

sysConfigCommand.....	194, 195
sysConfigComputerEthernetSpeed	162
sysConfigDhcpWait.....	154
sysConfigDhcpWaitDelay	154
sysConfigDownloadConfigFile	196, 198, 201
sysConfigDownloadConfigMode	201
sysConfigNetworkEthernetSpeed	162
sysConfigProductNamePadding	154
sysConfigStatsBySyslogEnable	346
sysConfigStatsNumberPeriods	346
sysConfigStatsPeriodLength.....	345
syslogConfigSource	21
syslogDhcpSiteSpecificCode	150, 342
syslogHost	20, 150, 342
syslogMsgDisplayLocalHost	343
syslogMsgDisplayMacAddress	343
syslogMsgDisplayTime	343
syslogMsgMaxSeverity	341, 354
syslogPort	20, 150, 342
syslogSelectConfigSource	150, 342
syslogStaticHost	342
syslogStaticPort	343
sysMacAddress.....	10, 155
sysMibVersion.....	140

T

telephonyAttributesAutomaticCallEnable	323
telephonyAttributesAutomaticCallTargetAddress	323
telephonyAttributesAutomaticRejection	329
telephonyAttributesCallDirectionRestriction	324
telephonyAttributesDelayedHotLineEnable	328
telephonyAttributesDelayedHotLineExtension	329
telephonyAttributesDelayedHotLineTargetAddress	329
telephonyAttributesHookFlashProcessing	324, 328
telephonyAttributesIpAddressCallEnable.....	325
telephonyCountrySelection	179

V

voicelfAdaptativeJitterBufferEnable	352
voicelfCodecG723Enable	234
voicelfCodecG723MaxPTime	235
voicelfCodecG723MinPTime	235
voicelfCodecG72616kbpsEnable.....	234
voicelfCodecG72616kbpsMaxPTime.....	235
voicelfCodecG72616kbpsMinPTime.....	235
voicelfCodecG72616kbpsPayloadType	234
voicelfCodecG72624kbpsEnable.....	234
voicelfCodecG72624kbpsMaxPTime.....	236
voicelfCodecG72624kbpsMinPTime.....	236
voicelfCodecG72624kbpsPayloadType	234
voicelfCodecG72632kbpsEnable.....	234
voicelfCodecG72632kbpsMaxPTime.....	236
voicelfCodecG72632kbpsMinPTime.....	236
voicelfCodecG72632kbpsPayloadType	234
voicelfCodecG72640kbpsEnable.....	234
voicelfCodecG72640kbpsMaxPTime.....	236
voicelfCodecG72640kbpsMinPTime.....	236
voicelfCodecG72640kbpsPayloadType	234
voicelfCodecG729Enable	234

voicelfCodecG729MaxPTime	236
voicelfCodecG729MinPTime	236
voicelfCodecPcmaEnable	233
voicelfCodecPcmaMaxPTime	235
voicelfCodecPcmaMinPTime	235
voicelfCodecPcmuEnable	233
voicelfCodecPcmuMaxPTime	235
voicelfCodecPcmuMinPTime	234
voicelfCodecPreferred	233
voicelfDtmfDetectionRiseTimeCriteria	242
voicelfDtmfDetectionUnitBreakPowerThreshold	242
voicelfDtmfDetectionUnitMaxPowerThreshold	241
voicelfDtmfDetectionUnitMinPowerThreshold	242
voicelfDtmfDetectionUnitNegativeTwist	242
voicelfDtmfDetectionUnitPositiveTwist	242
voicelfDtmfEnforceDefaultEvents	239
voicelfDtmfPayloadType	238
voicelfDtmfTransport	234, 237, 238, 239
voicelfEchoCancellationEnable	245
voicelfG711ComfortNoiseGenerationEnable	246
voicelfG711VoiceActivityDetectionEnable	244
voicelfG729VoiceActivityDetectionEnable	244
voicelfMaxJitterBufferLength	243, 352
voicelfSignalLimiterLevel	245
voicelfTargetJitterBufferLength	243, 352
voicelfUserInputGainOffset	246
voicelfUserOutputGainOffset	247

Index

Symbols

'+' character substitution in caller ID [280](#)

Numerics

10 BaseT [7](#), [11](#), [12](#), [35](#), [162](#)
 defined [419](#)
 see also *cabling*
100 BaseT [7](#), [11](#), [12](#), [35](#), [162](#)
 defined [419](#)
 see also *cabling*
802.1q, in QoS [117](#), [338](#)

A

acronyms [427](#)
ADSI
 caller ID [178](#)
 defined [419](#)
A-Law [79](#), [231](#)
 defined [419](#)
allow multiple active media in answer, in SIP [276](#)
analog lines
 see *lines*
 see *RJ21X connector*
analog modem, feature [79](#), [231](#)
area code, defined [419](#)
audience, intended [xvii](#)
authentication information [71](#), [262](#)
 request protection [263](#)
automatic
 call [103](#), [323](#)
 configuration update [46](#), [196](#)
 software update [58](#), [217](#)

B

Bootp BROADCAST flag, in DHCP requests [154](#)
branch matching method, in SIP [277](#)
branch, behaviour of Via in SIP [278](#)
broadcast storm, behaviour when restarting [22](#)
Bypass connection
 connecting [11](#), [12](#)
 defined [257](#)

C

cable length vs loop current [384](#)
cabling
 loop current vs cable length [384](#)
 RJ-11 [383](#)
 RJ21X
 creating [381](#)
 defined [381](#)
 Tip and Ring positions [381](#)
 RJ-45
 crossover [380](#)
 pin name [380](#)
 pinout information [379](#)
 straight through [379](#)
 see also *10 BaseT*
 see also *100 BaseT*

call
 automatic [103](#), [323](#)
 dialing sequence [303](#)
 emergency
 bypass [257](#)
 enabling [120](#), [304](#)
 forced SCN [304](#)
 forward
 on busy [91](#), [310](#)
 on no answer [93](#), [312](#)
 unconditional [94](#), [308](#)
 hold [102](#), [306](#)
 direction attributes [275](#)
 IP address [325](#)
 placing [4](#)
 putting on hold [99](#), [103](#), [307](#), [315](#)
 restriction on direction [324](#)
 second [103](#), [307](#)
 standard [303](#)
 transfer
 attended [96](#), [318](#)
 blind [96](#), [317](#)
 waiting [97](#), [314](#)
call forward
 on busy [92](#), [311](#)
 on no answer [94](#), [313](#)
 unconditional [95](#), [309](#)
call rejection [329](#)
call router, regular expression [229](#)
call transfer
 attended [97](#), [318](#)
 blind [97](#), [317](#)
call waiting
 controlling via SIP INFO [283](#)
 disabling [99](#), [315](#)
 enabling [99](#), [315](#)
 using [99](#), [315](#)
callee hangup supervision [227](#)
caller ID
 ADSI [178](#)
 blanking anonymous [228](#)
 country-specific, selecting [105](#), [179](#)
 DTMF signalling [177](#)
 ETSI 300 659-1 January 2001 (Annex B) [177](#)
 STD 220-250-713 Issue 01. November 1993 [177](#)
 TDK-TS 900 301-1 January 2003 [177](#)
 FSK generation [178](#)
 Bellcore GR-30-CORE [178](#)
 British Telecom (BT) SIN227, SIN242 [178](#)
 ETSI 300 659-1 [178](#)
 UK CCA specification TW/P&E/312 [178](#)
 generation [177](#)
CED tone, analog detection behaviour [250](#)
CED tone, detection [250](#)
clear channel fax
 enabling [88](#), [251](#)
 preferred codec [90](#), [251](#)
 setting [249](#)
CNG tone, detection [249](#)
codec
 data
 clear channel fax [249](#)
 enabling [88](#), [251](#)

- preferred 90, 251
 - enabling 249
 - T.38 88, 254, 255, 256
- voice
 - defined 231
 - DTMF detection 240
 - DTMF transport type 82, 83, 86, 237, 238
 - DTMF transport type over the SIP protocol 239
 - echo cancellation 83
 - enabling 84, 233
 - packetization time 84, 234
 - preferred 81, 233
- comfort noise 85, 246
- compliance to standards. *see standards compliance*
- configuration
 - file 141
 - using a GUI 137
 - web interface
 - configuration file download 40
 - enabling 25
 - Monitoring 29
 - password
 - modify 31
 - reset 32
 - SIP custom NAT traversal 123
 - STUN 122
 - user name, modify 32
 - web interface. *see web interface*
- configuration file
 - automatic update 46, 196
 - download server 40, 187
 - configuration source 41, 189
 - DHCP information, using 42, 189
 - HTTP server, configuring 41, 187
 - HTTPS server, configuring 188
 - IP address 42, 189
 - SNTP server, configuring 40, 187
 - static information, using 42, 190
 - TFTP server, configuring 40, 187
 - download, setting 42, 43, 190
 - encryption
 - decrypting generic 45, 193
 - decrypting specific 45, 193
 - defined 45, 193
 - example 204
 - HTTP, downloading via 46, 195
 - HTTPS, downloading via 195
 - management server, downloading from 201
 - syslog messages 44, 192
 - TFTP, downloading via 46, 194
 - user agent header, customizing, user agent header, customizing 195
 - web interface, configuring via 40
- configuration sources, setting all to static 145
- configuring the software
 - configuration file 141
 - DHCP (dynamic), using 144
 - MIB. *see SNMP*
 - static
 - setting configuration sources to 145
 - using 145
- connecting the unit 10
- connectors
 - analog lines (RJ21X) 7
 - Bypass 8
 - ETH1 7

- connectors (continued)
 - ETH2 7
 - FXS lines 7, 8
 - port numbering convention, FXS 8
 - reset / default 8
 - universal power supply unit 9
- country-specific parameters 385
 - caller ID, selecting 105, 179
 - setting 104, 178
- credential, in SIP 277
- crossover cable. *see cabling*
- custom tone configuration
 - SNMP, configuring via 180
 - web interface, configuring via 105
- customer services xxii

D

- default router, setting 37, 145
- Default Settings
 - factory reset procedure 21
 - disabling 22
 - in recovery mode 20
- delayed hot line 328
- DHCP information
 - Bootp BROADCAST flag 154
 - configuration file download server 42, 189
 - Image 53, 209
 - management server 335
 - network settings 36
 - options, waiting time to receive 153
 - SIP outbound proxy 64, 169
 - SIP Presence Compositor server 171
 - SIP proxy server 64, 167
 - SIP registrar server 64, 165
 - size of DHCP request 154
 - SNTP 38, 290
 - syslog daemon 29, 342
 - see also static information*
- DHCP server
 - configuring 155
 - defined 420
 - FQDN, entering 159
 - IP address of, setting 145
 - IP addresses, entering 158
 - network configuration 155
 - requirement 3
 - site specific option 157
 - vendor class ID 156
 - vendor specific option 156
- Dial Map. *see digit map*
- dialing
 - forced SCN call 304
 - IP address call 325
 - sequence 303
 - settings
 - DTMF duration value 240
 - inter-digit dial delay 240
 - standard call 303
 - telephone number 303
- Differentiated Services (DS) Field, in QoS 337
- digit map
 - # and * characters 75, 295
 - combining two expressions 74, 294
 - definition 73, 293

digit map (continued)

- examples
 - PBX emulation [300](#)
 - standard calls [298](#)
- processing digits behaviour [296](#)
- refused [297](#)
- rules [76](#), [296](#)
- special characters [74](#), [294](#)
- timeouts [298](#)
- timer [75](#), [295](#)
- using [74](#), [294](#)
- validating [75](#), [295](#)
- web interface, configuring via [73](#)

direction attribute "sendonly" [275](#)

direction attributes, in SIP [274](#), [275](#)

disabling lines [223](#)

distinctive ring [387](#)

DNS

- defined [420](#)
- primary [37](#), [145](#)
 - static [147](#)
- requirement [3](#)
- secondary [37](#), [145](#)
 - static [147](#)

DNS SRV

- call flow [174](#)
- defined [173](#)
- enabling [174](#)
- record lock, in SIP [175](#)

documentation

- Media5 download portal [xxi](#)
- Mediatrix download portal [xxi](#)

downgrading software, procedure [61](#), [220](#)

downloading software

- automatic update [58](#), [217](#)
- configuration source [52](#), [209](#)
- emergency download [61](#), [220](#)
- HTTP server, configuring [52](#), [207](#)
- HTTP, via [58](#), [216](#)
- HTTPS server, configuring [208](#)
- HTTPS, via [216](#)
- Image path [54](#), [211](#)
- LED states [57](#), [214](#)
- SNTP server, configuring [52](#), [207](#)
- Spanning Tree Protocol [61](#), [219](#)
- TFTP server, configuring [52](#), [207](#)
- TFTP, via [58](#), [215](#)
- troubleshooting [358](#)
- zip file [53](#), [211](#)

DTMF

- defined [420](#)
- detection [240](#)
- duration value [240](#)
- out-of-band [82](#), [237](#)
- RFC 2833 events [239](#)
- signalling, caller ID [177](#)
- transport type [82](#), [86](#), [237](#)
 - over the SIP protocol [239](#)
 - payload type [83](#), [238](#)
 - using SIP INFO method [238](#)
 - web interface, configuring via [82](#), [86](#)

E

echo cancellation [83](#), [245](#)

signal limiter [245](#)

emergency call

- bypass [257](#)
- enabling [120](#), [304](#)

emergency software download [61](#), [220](#)

enabling lines [223](#)

encryption, of configuration files

- decrypt generic [45](#), [193](#)
- decrypt specific [45](#), [193](#)
- defined [45](#), [193](#)

end user technical support [xxii](#)

Ethernet connection

- setting speed of [35](#), [162](#)
- web interface, configuring via [35](#)

examples of use

- MTU/MDU IP telephony [5](#)
- remote office connectivity [5](#)

F

factory reset

- disabling [22](#)
- reverting to [21](#)
- see also *recovery mode*

far end disconnect, signalling [226](#)

fax

- analog CED detection behaviour [250](#)
- call waiting tone, disabling [99](#)
- call waiting tone, disabling permanently [99](#), [316](#)
- call waiting tone, disabling temporarily [315](#)
- calling tone detection, enabling [249](#)
- CED tone detection, enabling [250](#)
- clear channel [249](#)
- T.38 [88](#), [254](#)
 - INVITE rejected with 606 [256](#)
 - no-signal [255](#)
 - user gain vs communication quality [246](#)

firmware download. see *software*

flash hook, setting [224](#)

Foreign Exchange Service/Station (FXS)

- defined [420](#)
- see also *lines*

FQDN, entering [159](#)

FSK generation, caller ID [178](#)

G

G.711 [79](#), [231](#)

- comfort noise [85](#), [246](#)
- defined [421](#)
- enabling [84](#)
- packetization time [84](#)
- voice activity detection [84](#), [244](#)

G.723.1 [80](#), [232](#)

- defined [421](#)
- enabling [85](#)
- packetization time [86](#)

G.726 [80](#), [232](#)

- comfort noise [246](#)
- defined [421](#)
- enabling [86](#)
- packetization time [87](#)
- payload type [86](#), [234](#)
- voice activity detection [244](#)

G.729 [80](#), [232](#)

- defined [421](#)
- enabling [85](#)

G.729 80, 232 (continued)
 packetization time 85
 voice activity detection 85, 244
GUI, using a 137

H

hardware
 cleaning 10
 condensation 10
 front indicators 7
 proper location 9
 rear connections 9
header, SIP user agent
 customizing 195, 260
 sending 260
hold, putting a call on 99, 102, 103, 306, 307, 315
 direction attributes 275
home domain proxy override 65, 259
hook flash processing 324
HTTP
 configuration file download 46, 195
 server
 configuring 41, 52, 187, 207
 requirement 3
 software download via 58, 216
HTTPS
 configuration file download 195
 server, configuring 188, 208
 software download via 216
humidity level 10

I

IEEE 802.1q, in QoS 117, 338
ignore username parameter, in SIP 283
Image server
 DHCP information, using 53, 209
 static information, using 53, 210
indicators of the hardware 7
installation
 connecting the hardware 10
 package contents 3
 provisioning sequence, initial 14
 rack-mounting 10
 requirements 3
 reserving IP address 10
 RJ21X connector 381
 safety recommendations 3
 selecting site for 9
 setting up the unit for the first time 13
 verifying 23
intended audience xvii
inter-digit dial delay 240
international code mapping 280
INVITE rejected with 606, in T.38 256
IP address
 default router 37, 145
 defining
 decimal 143
 hexadecimal 143
 octal 143
 DHCP server 145
 DHCP, using 144
 DNS, primary 37, 145
 static 147

IP address (continued)
 DNS, secondary 37, 145
 static 147
 download server 42, 189
 entering 158
 ETH2 connector, static 153
 Image server 52, 209
 locating 143
 Management Server 335
 of unit 37, 145
 SIP outbound proxy 65, 169
 SIP Presence Compositor server 65
 SIP proxy server 65, 167, 171
 SIP registrar server 65, 165
 SNTP server 290
 static
 setting configuration sources to 145
 using 145
 subnet mask 37, 145
 syslog 30
 syslog daemon 342
 vocal identification of 15
IP address call 325
IP address, dialing 325

J

jitter
 buffer protection 82, 243
 defined 421
 web interface, configuring via 82

L

LAN
 cable 23
 defined 422
LEDs
 behaviour
 in download mode 57, 214
 in starting mode 15
 defined 422
 In Use 15
 LAN 15
 patterns
 AdminMode 16, 18
 Booting 16, 17
 DefaultSettings ending 16
 DiagFailed 17
 ImageDownloadError 16
 ImageDownloadInProgress 16
 InitFailed 17
 NormalMode 16, 17
 RebootPending 16
 RecoveryMode 16, 18
 RecoveryModePending 16
 ResetPending 16
 Power 16
 Ready 15
 states 15
line mapping 225
lines
 anonymous caller ID, blanking of 228
 callee hangup supervision 227
 comfort noise 85, 246

lines (continued)

- data codecs [87](#), [249](#)
 - analog CED detection behaviour [250](#)
 - CED fax tone detection [250](#)
 - clear channel fax [249](#)
 - clear channel, enabling [88](#), [251](#)
 - clear channel, preferred [90](#), [251](#)
 - enabling [249](#)
 - fax calling tone detection [249](#)
 - T.38 [88](#), [254](#), [255](#), [256](#)
- echo cancellation [245](#)
 - signal limiter [245](#)
- far end disconnect, signalling [226](#)
- jitter buffer protection [82](#), [243](#)
- locking/unlocking [223](#)
- reversal [228](#)
- source selection [225](#)
 - FXS to FXO line mapping [225](#)
 - reserving FXS line [226](#)
- unregistered, behaviour when [64](#), [224](#)
- user gain [246](#)
- voice activity detection [84](#), [244](#)
- voice codecs
 - defined [231](#)
 - DTMF detection [240](#)
 - DTMF transport type [82](#), [83](#), [86](#), [237](#), [238](#), [239](#)
 - echo cancellation [83](#)
 - enabling [84](#), [233](#)
 - packetization time [84](#), [234](#)
 - preferred codec [81](#), [233](#)

local

- host, in customized syslog messages [343](#)
- IP address, setting [37](#), [145](#)
- ring behaviour, in SIP [276](#)
- time, in customized syslog messages [343](#)

location

- caller ID, selecting [105](#), [179](#)
- country, setting [104](#), [178](#)
- web interface, configuring via [104](#)

locking lines [223](#)loop current vs cable length [384](#)loop current, setting [226](#)**M**MAC address [10](#)

- defined [422](#)
- in customized syslog messages [343](#)
- vocal identification of [15](#)

making

- forced SCN call [304](#)
- IP address call [325](#)
- standard call [303](#)

Management Server

- defined [422](#)
- DHCP information, using [335](#)
- in configuration file download [201](#)
- static information, using [336](#)
- using [335](#)

Max-Forwards header, in SIP [273](#)Media5 download portal [xxi](#)Mediatix download portal [xxi](#)

message waiting indicator

- defined [112](#), [331](#)
- notifications [332](#)
- Notify service [114](#), [333](#)

message waiting indicator (continued)

- refresh subscription [332](#)
- setting up [113](#), [331](#)
- web interface, configuring via [112](#)

MIB

- defined [127](#)
- in SNMP protocol [127](#)
 - see also *parameters*

MIB structure

- changing a parameter value [137](#)
- conformance [136](#)
- description
 - mediatrixAdmin [135](#)
 - mediatrixConfig [135](#)
 - mediatrixExperimental [135](#)
 - mediatrixIpTelephonySignaling [135](#)
 - mediatrixMgmt [135](#)
 - mediatrixModules [135](#)
 - mediatrixProducts [135](#)

events [136](#)introduction [127](#)objects [136](#)OID, defined [127](#)persistent parameters [137](#)SMI, defined [127](#)SNMP messages. see *SNMP*

tables

- defined [138](#)
- generic [138](#)
- groupAdmin [139](#)
- ifAdmin [139](#)
- textual conventions [136](#)
- volatile parameters [137](#)

mounting, on a rack [10](#)MTU, requirements [349](#)MTU/MDU IP telephony, example of use [5](#)Mu (μ)-Law [79](#), [231](#)defined [422](#)**N**

NAT

- in configuration file download [46](#), [194](#)
- traversal, setting IP address of [264](#)
- see also *STUN*

network settings

- DHCP information, using [36](#)
- static information, using [36](#)

no-signal, in T.38 [255](#)**O**Offer/Answer model [284](#)

- allow audio and image negotiation [285](#)
- allow media reactivation in answer [285](#)
- codec order in answer [286](#)

OID, defined [127](#)operating temperature [10](#)outbound proxy usage with Route header [280](#)out-of-band DTMF [82](#), [237](#)overview of the product [4](#)**P**package contents [3](#)packetization time, setting for voice codecs [84](#), [234](#)

- parameters
 - changing value of [137](#)
 - using a GUI [137](#)
- payload type
 - asymmetric [282](#)
 - using the one found in answer [282](#)
- PCM traces, enabling [353](#)
- penalty box, in SIP [67](#), [267](#)
- persistent parameters, defined [137](#)
- PIN dialing, defined [326](#)
- placing a call [4](#)
- port number
 - configuration file fetching [150](#)
 - configuration file server [42](#), [189](#)
 - DHCP setting [158](#)
 - image server [52](#), [148](#), [209](#)
 - management server [149](#), [335](#)
 - RTCP, setting range of [83](#)
 - RTP, setting range of [83](#)
 - SIP [64](#), [66](#), [259](#)
 - SIP outbound proxy [65](#), [151](#), [168](#)
 - SIP Presence Composer [65](#), [171](#)
 - SIP proxy [65](#), [151](#), [167](#)
 - SIP registrar [65](#), [151](#), [165](#)
 - SNMP agent [147](#)
 - SNMP trap [147](#)
 - syslog [30](#), [150](#), [342](#)
- port numbering convention, FXS connectors [8](#)
- primary DNS, setting [37](#)
- product overview [4](#)
- provisioning
 - configuration file [141](#)
 - initial sequence [14](#)
 - MIB files [141](#)
- publications expiration, in SIP [270](#)
- publications refresh, in SIP [269](#)
 - via web interface [67](#)

Q

- QoS
 - 802.1q [117](#), [338](#)
 - defined [423](#)
 - Differentiated Services (DS) Field [337](#)
 - VLAN [339](#)
 - voice QoS vs RTCP packets [339](#)
 - web interface, configuring via [117](#)

R

- rack-mounting the unit [10](#)
- rear connections [9](#)
- recovery mode
 - LED patterns [18](#)
 - resetting in [20](#)
 - see also *factory reset*
- Referred-By field, in SIP [274](#)
- registration expiration default, in SIP [269](#), [270](#)
- registration expiration, in SIP [268](#)
- registration refresh, in SIP [268](#)
 - via web interface [66](#)
- regular expression, defined [229](#)
- related documentation [xviii](#)
- remote line extension [327](#)
- remote office connectivity, example of use [5](#)

- Replaces header, in SIP
 - configuration [272](#)
 - version [272](#)
- requirements [3](#)
- restart
 - behaviour in case of broadcast storm [22](#)
 - software-initiated [22](#)
 - unit [28](#)
- reversal, of a line [228](#)
- RFC
 - RFC 1350 [373](#)
 - RFC 1542 [154](#)
 - RFC 1769 [37](#), [289](#)
 - RFC 1889 [373](#)
 - RFC 1890 [81](#), [238](#), [373](#)
 - RFC 1945 [373](#)
 - RFC 2131 [155](#), [373](#)
 - RFC 2132 [155](#), [156](#), [157](#), [373](#)
 - RFC 2246 [188](#), [208](#)
 - RFC 2347 [373](#)
 - RFC 2348 [373](#)
 - RFC 2349 [373](#)
 - RFC 2459 [188](#), [208](#)
 - RFC 2475 [119](#), [337](#)
 - RFC 2543 [170](#), [274](#), [277](#)
 - RFC 2543bis [273](#)
 - RFC 2616 [25](#)
 - RFC 2617 [31](#), [373](#)
 - RFC 2705 [73](#), [293](#)
 - RFC 2782 [173](#)
 - RFC 2818 [188](#), [208](#)
 - RFC 2833 [81](#), [237](#), [239](#), [373](#)
 - RFC 2976 [238](#), [324](#)
 - RFC 3164 [29](#), [341](#), [373](#)
 - RFC 3261 [68](#), [71](#), [170](#), [260](#), [262](#), [265](#), [269](#), [273](#), [277](#), [373](#)
 - RFC 3263 [173](#)
 - RFC 3264 [274](#), [275](#), [284](#)
 - RFC 3280 [188](#), [208](#)
 - RFC 3389 [246](#), [373](#)
 - RFC 3489 [121](#), [287](#), [373](#)
 - RFC 3863 [63](#), [171](#), [269](#)
 - RFC 3903 [63](#), [171](#), [269](#)
 - RFC 4579 [319](#)
- ring, distinctive [387](#)
- ringing response code, in SIP [279](#)
- RJ-11. see *cabling*
- RJ21X connector
 - creating [381](#)
 - defined [381](#)
 - Tip and Ring positions [381](#)
- RJ-45. see *cabling*
- RTCP [117](#), [337](#)
 - base port range [83](#)
- RTP, base port range [83](#)

S

- safety
 - recommendations [3](#), [371](#)
 - warnings
 - Circuit Breaker (15A) [370](#)
 - LAN Connector [370](#)
 - No. 26 AWG [370](#)
 - Product Disposal [370](#)
 - Socket Outlet [370](#)
 - TN Power [370](#)

- second call, service [103](#), [307](#)
- secondary DNS, setting [37](#)
- session timer
 - enabling [261](#)
 - session expiration delay
 - maximum [261](#)
 - minimum [261](#)
 - version supported, setting [261](#)
- signal limiter [245](#)
- signalling protocol, SIP. see *SIP, setting*
- SIP INFO, controlling call waiting tone via [283](#)
- SIP, setting
 - allow audio and image negotiation, web interface, configuring via [70](#)
 - allow media reactivation in answer, web interface, configuring via [70](#)
 - allow multiple active media in answer [276](#)
 - branch matching method [277](#)
 - call waiting tone via SIP INFO [283](#)
 - configuration [259](#)
 - credential [277](#)
 - direction attribute "sendonly" [275](#)
 - direction attributes [275](#)
 - direction attributes present [274](#)
 - DNS SRV record lock [175](#)
 - escape pound (#) character in SIP URI username [284](#)
 - web interface, configuring via [69](#)
 - failed registration attempts [281](#)
 - from URI content [281](#)
 - home domain in Request URI [281](#)
 - home domain override [260](#)
 - home domain proxy override [65](#), [259](#)
 - ignore username parameter [283](#)
 - international code mapping [280](#)
 - local ring behaviour [276](#)
 - Max-Forwards header [273](#)
 - message waiting indicator notifications [332](#)
 - NAT traversal [264](#)
 - network asserted caller ID [282](#)
 - Offer/Answer model [284](#)
 - allow audio and image negotiation [285](#)
 - allow media reactivation in answer [285](#)
 - codec order in answer [286](#)
 - outbound proxy
 - DHCP information, using [64](#), [169](#)
 - loose router status [170](#)
 - static information, using [64](#), [169](#)
 - usage with Route header [280](#)
 - outbound proxy server, web interface, configuring via [64](#)
 - payload type in answer, using [282](#)
 - payload type, asymmetric [282](#)
 - penalty box [267](#)
 - web interface, configuring via [67](#)
 - Presence Compositor server
 - DHCP information, using [171](#)
 - static information, using [172](#)
 - proxy server
 - DHCP information, using [64](#), [167](#)
 - static information, using [64](#), [168](#)
 - web interface, configuring via [64](#)
 - publications expiration [270](#)
 - publications refresh [269](#)
 - web interface, configuring via [67](#)
 - Referred-By field [274](#)
 - registrar server
 - DHCP information, using [64](#), [165](#)
 - static information, using [64](#), [166](#)
 - SIP, setting (continued)
 - registrar server
 - web interface, configuring via [64](#)
 - registration expiration [268](#)
 - registration expiration, default [269](#), [270](#)
 - registration refresh [268](#)
 - web interface, configuring via [66](#)
 - replaces
 - configuration [272](#)
 - version [272](#)
 - ringing response code [279](#)
 - session timer [261](#)
 - session expiration delay, maximum [261](#)
 - session expiration delay, minimum [261](#)
 - version supported [261](#)
 - T.38 negotiation syntax [280](#)
 - transmission timeout [273](#)
 - transport type
 - TCP [68](#), [265](#)
 - UDP [68](#), [265](#)
 - web interface, configuring via [68](#)
 - trusted sources [264](#)
 - UDP checksum [286](#)
 - UDP source port behaviour [266](#)
 - unsupported INFO request [279](#)
 - uri-parameters [279](#)
 - user agents
 - authentication information [262](#), [263](#)
 - authentication information [71](#)
 - display name [260](#)
 - header, customizing [195](#), [260](#)
 - header, enabling to send [260](#)
 - main user name [259](#)
 - other accepted user names [260](#)
 - setting information [66](#), [259](#)
 - web interface, configuring via [65](#)
 - Via branch behaviour [278](#)
 - site specific information, DHCP setting [157](#)
 - site, selecting for unit [9](#)
 - SMI, defined [127](#)
 - SNMP
 - access limitation [140](#)
 - behaviour [129](#)
 - non-secure management mode [130](#)
 - secure management mode [130](#)
 - configuring [147](#)
 - custom tone configuration [180](#)
 - messages [128](#)
 - MIB [127](#)
 - service agent [140](#)
 - SNMP configuration file [131](#)
 - versions [128](#)
 - SNTP
 - defined [424](#)
 - DHCP information, using [38](#), [290](#)
 - enabling [37](#), [289](#)
 - server, configuring [40](#), [52](#), [187](#), [207](#)
 - static information, using [38](#), [290](#)
 - time zone, defining custom [37](#), [291](#)
 - web interface, configuring via [37](#)
 - software
 - downgrading, procedure [61](#), [220](#)
 - downloading
 - automatic update [58](#), [217](#)
 - configuration source [52](#), [209](#)
 - emergency procedure [61](#), [220](#)

software (continued)

downloading

- HTTP server, configuring [52](#), [207](#)
- HTTP, via [58](#), [216](#)
- HTTPS server, configuring [208](#)
- HTTPS, via [216](#)
- Image path [54](#), [211](#)
- LED states [57](#), [214](#)
- SNTP server, configuring [52](#), [207](#)
- Spanning Tree Protocol [61](#), [219](#)
- syslog messages [56](#), [213](#)
- TFTP server, configuring [52](#), [207](#)
- TFTP, via [58](#), [215](#)
- troubleshooting [358](#)
- user agent header, customizing, user agent header, customizing [216](#)
- zip file [53](#), [211](#)
- see also *downgrading*

source line selection [225](#)

- FXS to FXO line mapping [225](#)
- reserving FXS line [226](#)

Spanning Tree Protocol [61](#), [219](#)

- DHCP options waiting time [153](#)

special vocal features [15](#)

- IP address [15](#)
- MAC address [15](#)

standards compliance

- agency approvals [365](#)
- CE marking [367](#)
- emissions [365](#)
- FCC Part 15 disclaimer [366](#)
- FCC Part 68 disclaimer [366](#)
- immunity [365](#)
- Industry Canada disclaimer [367](#)
- RoHS China [368](#)
- safety standards [365](#)
- telecom [366](#)

standards supported xxi

caller ID

- Bellcore GR-30-CORE [178](#)
- British Telecom (BT) SIN227, SIN242 [178](#)
- ETSI 300 659-1 [178](#)
- ETSI 300 659-1 January 2001 (Annex B) [177](#)
- STD 220-250-713 Issue 01. November 1993 [177](#)
- TDK-TS 900 301-1 January 2003 [177](#)
- UK CCA specification TW/P&E/312 [178](#)
- draft-choudhuri-sip-info-digit-00.txt [81](#), [237](#), [238](#), [239](#), [326](#)
- draft-ietf-sip-cc-transfer-02.txt [271](#)
- draft-ietf-sip-cc-transfer-05.txt [271](#)
- draft-ietf-sipping-cc-transfer-01.txt [271](#)
- draft-ietf-sipping-mwi-01.txt [113](#), [331](#)
- draft-ietf-sipping-realtimefax-01.txt [88](#), [254](#)
- draft-ietf-sip-refer-02.txt [271](#)
- draft-ietf-sip-session-timer-04.txt [261](#)
- draft-ietf-sip-session-timer-08.txt [261](#)
- draft-mahy-sip-message-waiting-02.txt [114](#), [333](#)
- GR-506-CORE (Issue 1, with Revision 1, November 1996) [113](#), [331](#)
- ITU-T Q.24 [81](#), [237](#)
- Recommendation ITU T.38 version 0 [88](#), [254](#)
- Recommendation ITU-T T.38, section D.2.3 [280](#)
- RFC 1350 [373](#)
- RFC 1542 [154](#)
- RFC 1769 [37](#), [289](#)
- RFC 1889 [373](#)
- RFC 1890 [81](#), [238](#), [373](#)

standards supported xxi (continued)

- RFC 1945 [373](#)
- RFC 2131 [155](#), [373](#)
- RFC 2132 [155](#), [156](#), [157](#), [373](#)
- RFC 2246 [188](#), [208](#)
- RFC 2347 [373](#)
- RFC 2348 [373](#)
- RFC 2349 [373](#)
- RFC 2459 [188](#), [208](#)
- RFC 2475 [119](#), [337](#)
- RFC 2543 [170](#), [274](#), [277](#)
- RFC 2543bis [273](#)
- RFC 2616 [25](#)
- RFC 2617 [31](#), [373](#)
- RFC 2705 [73](#), [293](#)
- RFC 2782 [173](#)
- RFC 2818 [188](#), [208](#)
- RFC 2833 [81](#), [237](#), [239](#), [373](#)
- RFC 2976 [238](#), [324](#)
- RFC 3164 [29](#), [341](#), [373](#)
- RFC 3261 [68](#), [71](#), [170](#), [260](#), [262](#), [265](#), [269](#), [273](#), [277](#), [373](#)
- RFC 3263 [173](#)
- RFC 3264 [274](#), [275](#), [284](#)
- RFC 3280 [188](#), [208](#)
- RFC 3389 [246](#), [373](#)
- RFC 3489 [121](#), [287](#), [373](#)
- RFC 3515 [271](#)
- RFC 3863 [63](#), [171](#), [269](#)
- RFC 3903 [63](#), [171](#), [269](#)
- RFC 4579 [319](#)
- sip-replaces-01 draft [272](#)
- sip-replaces-03 draft [272](#)
- Telecordia GR-1401-CORE (Issue 1, June 2000) [113](#), [331](#)

static

- ETH2 connector IP address [153](#)
- setting DNS address as [147](#)

static information

- configuration file download server [42](#), [190](#)
- Image [53](#), [210](#)
- management server [336](#)
- network settings [36](#)
- setting all configuration sources to [145](#)
- SIP outbound proxy [64](#), [169](#)
- SIP Presence Compositor server [172](#)
- SIP proxy server [64](#), [168](#)
- SIP registrar server [64](#), [166](#)
- SNTP [38](#), [290](#)
- syslog daemon [29](#), [342](#)
- see also *DHCP information*

statistics

- by syslog [346](#)
- resetting [346](#)
- RTP [345](#)
- setting how to collect [345](#)
- viewing [345](#)

straight through cable. see *cabling*

STUN

- configuring [288](#)
- defined [121](#), [287](#)
- web interface, configuring [122](#)

subnet mask, setting [37](#), [145](#)

subscriber services

- call forward
 - on busy [91](#), [310](#)
 - on no answer [93](#), [312](#)
 - unconditional [94](#), [308](#)

subscriber services (continued)

- call forward
 - web interface, configuring via [90](#)
- call transfer
 - attended transfer [96](#), [318](#)
 - blind transfer [96](#), [317](#)
 - web interface, configuring via [96](#)
- call waiting [97](#), [314](#)
 - web interface, configuring via [97](#)
- conference call [100](#), [319](#)
 - web interface, configuring via [100](#)
- hold [102](#), [306](#)
 - direction attributes [275](#)
 - web interface, configuring via [102](#)
- second call [103](#), [307](#)
 - web interface, configuring via [103](#)
- service activation processing [305](#)

support services [xxii](#)

syslog

- daemon
 - configuration via web interface [29](#)
 - configuring the application [30](#), [343](#)
 - customizing messages
 - local host [343](#)
 - local time [343](#)
 - MAC address [343](#)
 - defined [29](#), [341](#)
 - DHCP information, using [29](#), [342](#)
 - enabling [341](#)
 - messages examples [30](#), [341](#)
 - requirement [3](#)
 - static information, using [29](#), [342](#)
 - statistics [346](#)
- messages
 - configuration file [44](#), [192](#)
 - examples [30](#), [341](#)
 - software download [56](#), [213](#)

T

T.38

- defined [424](#)
- enabling [89](#), [254](#)
- negotiation syntax [280](#)
- not supported by other endpoint [355](#)
- number of redundancy packets [254](#)
- protection level [254](#)
- redundancy parameters [89](#)

tables

- defined [138](#)
- generic [138](#)
- groupAdmin [139](#)
- ifAdmin [139](#)

TCP, transport type [68](#), [265](#)technical support for end user [xxii](#)telephone number, dialing [303](#)

telephony attributes

- automatic call [103](#), [323](#)
 - web interface, configuring via [103](#)
- call direction restriction [324](#)
- hook flash processing [324](#)
- IP address call [325](#)

temperature, operating [10](#)textual conventions, in MIB structure [136](#)

TFTP

- configuration file download [46](#), [194](#)

TFTP (continued)

server

- configuring [40](#), [52](#), [187](#), [207](#)
- defined [424](#)
- requirement [3](#)
- software download via [58](#), [215](#)

time zone, defining custom [37](#), [291](#)TPE. see *cabling*transfer, version supported, setting [271](#)

transferring a call

- attended transfer [97](#), [318](#)
- blind [97](#), [317](#)

translated warning definition [369](#)transmission timeout, setting [273](#)

troubleshooting

- bypass feature does not activate [352](#)
- call
 - 3-way conference lost conversation [355](#)
 - cannot establish to endpoint [355](#)
 - cannot make [354](#)
 - cannot make or receive [354](#)
 - cannot disable adaptive jitter buffer [352](#)
 - configuration, configuration source does not work [358](#)
 - DHCP unreachable [352](#)
- fax
 - poor line condition during transmission [355](#)
 - specific issues [357](#)
 - T.38 transmission fails [356](#)
 - tested fax models [356](#)
 - unable to send in clear channel [355](#)
 - unable to send in T.38 [355](#)
 - voice does not switch back to codec after clear channel fax [356](#)

LEDs, all off [351](#)long delay when starting unit [351](#)PCM traces, enabling [353](#)

SNMP

- cannot set a variable [360](#), [361](#)
- network manager cannot access unit [360](#)
- no response when trying to access unit [360](#)
- SNMPv3 variables contents [361](#)
- traps not received by network manager [360](#)
- when viewing table, unit does not respond [361](#)
- wrong value error message [361](#)
- software download
 - cannot communicate with image server [358](#)
 - downgrade fails [359](#)
 - long time to perform [359](#)
 - path not recognized [359](#)
 - transfer problems [359](#)
- unable to reach unit after changing Ethernet speed [352](#)
- value not accepted [352](#)

trusted sources, in SIP [264](#)**U**

UDP

- checksum [286](#)
- source port behaviour [266](#)
- transport type [68](#), [265](#)

unit, restarting [28](#)

Unit Manager Network product

- as management server [3](#), [147](#), [149](#), [335](#)
- defined [xxii](#)
- using [51](#), [137](#), [143](#), [207](#)

unlocking lines [223](#)

unsupported INFO request, in SIP [279](#)
uri-parameters, in SIP [279](#)
using this manual [xx](#)
UTP. see *cabling*

V

vendor specific information, DHCP setting [156](#)
verifying the installation [23](#)
viewing statistics and performances [345](#)
VLAN, in QoS [339](#)
vocal features, special [15](#)
 IP address [15](#)
 MAC address [15](#)
voice activity detection [84](#), [244](#)
volatile parameters, defined [137](#)

W

web interface
 access limitation [25](#)
 allow audio and image negotiation [70](#)
 allow media reactivation in answer [70](#)
 automatic call [103](#)
 call forward [90](#)
 call hold [102](#)
 call transfer [96](#)
 call waiting [97](#)
 choosing suitable web browser [25](#)
 codecs [79](#)
 conference call [100](#)
 configuration file download [40](#)
 country [104](#)
 custom tone configuration [105](#)
 default router [37](#)
 digit map [73](#)
 DTMF transport type [82](#), [86](#)
 enabling [25](#)
 escape pound (#) character in SIP URI username [69](#)
 Ethernet connection speed [35](#)
 firmware download [51](#)
 group port management [34](#)
 interface management [34](#)
 jitter [82](#)
 local IP address [37](#)
 message waiting indicator [112](#)
 Monitoring [29](#)
 password
 modify [31](#)
 reset [32](#)
 primary DNS [37](#)
 QoS [117](#)
 second call [103](#)
 secondary DNS [37](#)
 SIP custom NAT traversal [123](#)
 SIP outbound proxy server [64](#)
 SIP penalty box [67](#)
 SIP proxy server [64](#)
 SIP publications refresh [67](#)
 SIP registrar server [64](#)
 SIP registration refresh [66](#)
 SIP transport type [68](#)
 SIP user agents [65](#)
 SNTP [37](#)
 status, system [27](#)
 STUN [122](#)

web interface (continued)
 subnet mask [37](#)
 system management [33](#)
 user name, modify [32](#)