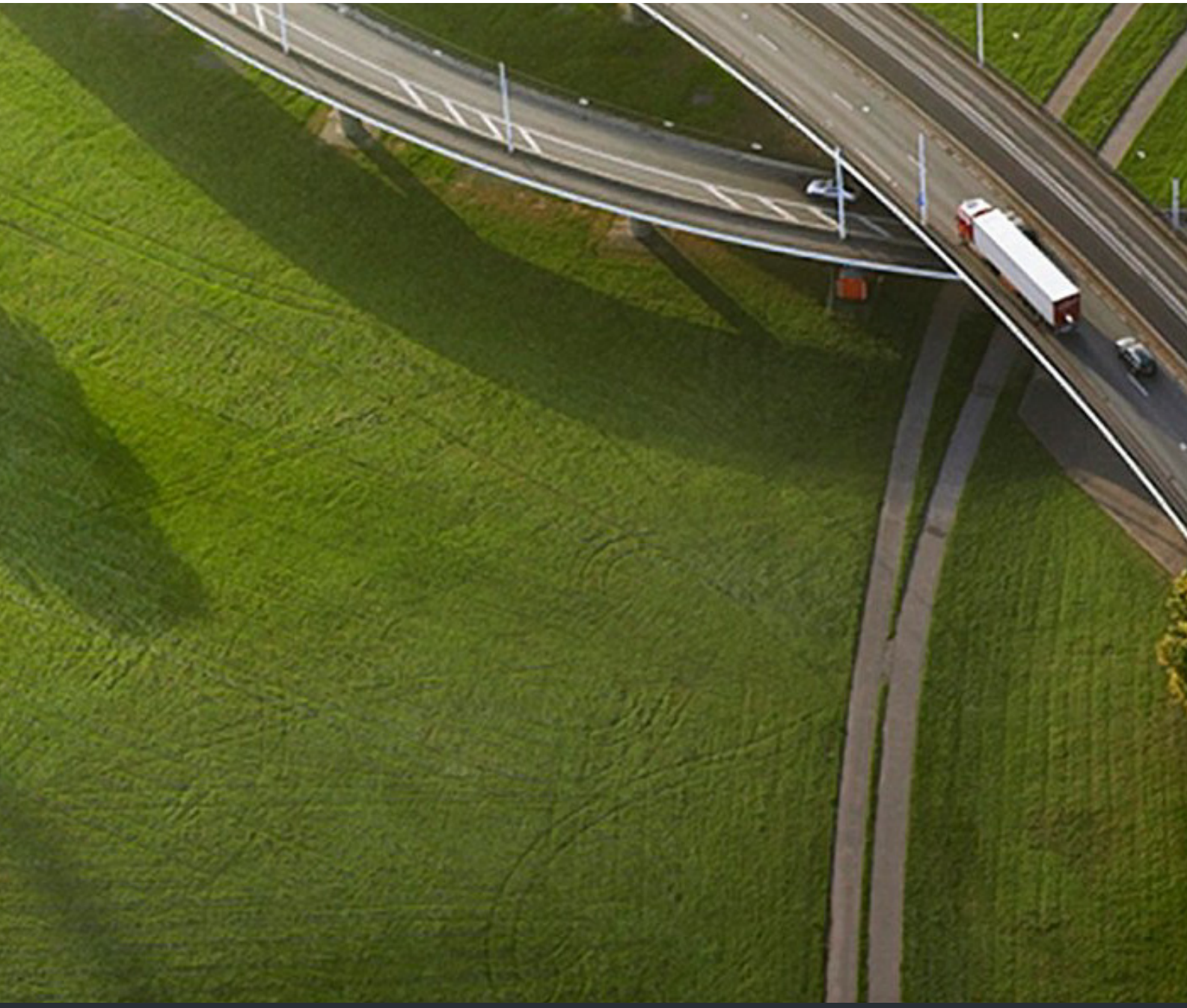
UNIFY Harmonize your enterprise



OpenScape Voice V7
Volume 2: SOAP/XML Subscriber Interface Provisioning

Interface Manual

A31003-H8070-T101-11-7618

Our Quality and Environmental Management Systems are implemented according to the requirements of the ISO9001 and ISO14001 standards and are certified by an external certification company.

Copyright © Unify Software and Solutions GmbH & Co. KG 6/2015
Mies-van-der-Rohe-Str. 6, 80807 Munich/Germany

All rights reserved.

Reference No.: A31003-H8070-T101-11-7618

The information provided in this document contains merely general descriptions or characteristics of performance which in case of actual use do not always apply as described or which may change as a result of further development of the products.
An obligation to provide the respective characteristics shall only exist if expressly agreed in the terms of contract.

Availability and technical specifications are subject to change without notice.

Unify, OpenScape, OpenStage and HiPath are registered trademarks of Unify Software and Solutions GmbH & Co. KG. All other company, brand, product and service names are trademarks or registered trademarks of their respective holders.

unify.com

Harmonize
your enterprise

Contents

1 Introduction . . . 23
1.1 Deployment Network Configuration . . . 23
1.2 Subscriber Interactions . . . 23
1.2.1 Service Interactions . . . 24
1.3 Subscriber Provisioning Network Architecture Layout . . . 24
1.3.1 Small Medium Enterprise . . . 25
1.3.2 OpenScape Voice Assistant . . . 25
1.3.3 SIP Network Servers . . . 25

2 Changes Between Version V6 and V7 . . . 26

3 Design Considerations . . . 31
3.1 Provisioning Traffic Volumes . . . 31
3.2 Sizing . . . 31
3.3 System Requirements . . . 31
3.4 Command Line Interface (CLI) Access . . . 31
3.5 Security . . . 31
3.6 Requirements for Using this Interface . . . 32
3.6.1 CRM Required Features . . . 32
3.6.2 Required Interface Support . . . 33
3.7 Standards Support . . . 33
3.8 Interface Version Support . . . 33

4 Concepts . . . 34
4.1 Subscribers . . . 34
4.1.1 Profile Only Subscribers . . . 34
4.2 Feature Profiles . . . 34
4.3 Branch Offices . . . 36
4.4 Recent Change Logging . . . 37
4.4.1 Recent Change Logging in the SOAP Server . . . 38
4.4.2 Recent Change Logging in SOAP Mass Provisioning and SOAP Export . . . 40
4.5 Hunt Groups . . . 40
4.5.1 Types of Hunt Groups . . . 41
4.5.2 Hunt Algorithms . . . 41
4.5.3 Blocking of DNs . . . 42
4.5.4 Traffic Measurements . . . 42
4.5.5 Hunt Group Restrictions . . . 44
4.5.6 Hunt Group Traffic Measurements . . . 44
4.6 Jobs . . . 45
4.7 Media Gateways and Servers . . . 45
4.8 Calling Number Display Modification . . . 46
4.8.1 Display Number Definition Table . . . 47
4.8.2 Display Number Prefix Table . . . 48
4.8.3 Display Number Modification Table . . . 50
4.9 Business Groups . . . 52
4.10 Business Group Department and Services . . . 52
4.10.1 Main Numbers . . . 52
4.10.2 Attendant Numbers . . . 53

4.10.3 Message Detail Recording (MDR) Information . . . 53
4.10.4 Business Group Subnet E911 LIN Administration . . . 53
4.11 Business Group Traffic Measurements . . . 54
4.12 Numbering Plan . . . 54
4.12.1 Service Activation Codes . . . 55
4.13 Keyset Information . . . 55
4.13.1 Keyset Use (enum) . . . 55
4.13.2 Manual Hold Timer (range 5..3600) . . . 56
4.13.3 Line Appearances . . . 56
4.13.3.1 Keyset Operation Mode . . . 57
4.13.4 Call Forward LED Indication . . . 57
4.14 Office Codes . . . 57
4.15 Home DNs . . . 58
4.16 SIP Endpoints . . . 58
4.17 EndPoint Profiles . . . 58
4.18 SIP Network Servers . . . 59
4.19 Dynamic CAC Router Management . . . 59

5 External Interfaces . . . 61
5.1 Interface (WSDL) Versions . . . 61
5.1.1 The Interface WSDL and Interface Release Control . . . 61
5.2 Obsolete Method Names . . . 62
5.3 SOAP / XML Conventions . . . 65
5.4 Primitive Data Definitions . . . 66
5.4.1 Field Data Type . . . 66
5.4.2 Field Definitions . . . 67
5.5 Paging . . . 72
5.6 Return Code Information . . . 73

6 Security on OpenScape Voice and the OS Voice SOAP Server Interface . . . 74
6.1 Use of TLS with the OSV Soap Server . . . 75
6.1.1 Configuration of RTP Parameters . . . 75
6.1.2 Certificates and TLS Verification . . . 76
6.2 Create Client Profile . . . 80
6.3 Modify Client Profile . . . 82
6.4 Delete Client Profile . . . 84
6.5 Get Client Profile . . . 85
6.6 Get Client Profile List . . . 86
6.7 Create PFR (Packet Filter Rule) . . . 86
6.8 Modify PFR (Packet Filter Rule) . . . 88
6.9 Delete PFR (Packet Filter Rule) . . . 89
6.10 Query PFR (Packet Filter Rule) . . . 90
6.11 Get PFR Symbolic Name List . . . 92

7 General Utility Interfaces . . . 94
7.1 Get Version . . . 94
7.2 Get Feature List . . . 94
7.3 Get Config Data . . . 95
7.4 Set Config Parm Data . . . 96
7.5 Get Config Parm Data . . . 98
7.6 Get COS List . . . 99
7.7 Get Rate Area List . . . 99
7.8 Get Intercept List . . . 100

7.9 Get Calling Location List . . . 100
7.10 Get Associated SIP Endpoint List . . . 101
7.11 Get Licensing Info . . . 102
7.12 Get Licensing Usage Information . . . 107
7.13 Get Destination List . . . 109
7.14 Translate Dialed Digits . . . 110
7.15 Get Supported Language List . . . 112
7.16 Get Total Number of Call Pickup Groups . . . 113
7.17 Get Fully Qualified Number . . . 114
7.18 Get System Info . . . 115

8 Network Management Interfaces via SOAP . . . 117
8.1 Office Code . . . 117
8.1.1 Create Office Code . . . 117
8.1.2 Modify Office Code . . . 118
8.1.3 Delete Office Code . . . 119
8.1.4 Display Office Code . . . 119
8.2 Home DN . . . 121
8.2.1 Create Home Dn . . . 121
8.2.2 Modify Home Dn . . . 122
8.2.3 Delete Home Dn . . . 124
8.2.4 Query Home Dn . . . 125
8.3 Endpoint . . . 127
8.3.1 Create Endpoint . . . 127
8.3.2 Modify Endpoint . . . 139
8.3.3 Delete Endpoint . . . 151
8.3.4 Display Endpoint . . . 151
8.4 Destination . . . 153
8.4.1 Create Destination . . . 153
8.4.2 Modify Destination . . . 155
8.4.3 Delete Destination . . . 156
8.4.4 Display Destination . . . 157
8.5 Rate Area . . . 159
8.5.1 Create Rate Area . . . 159
8.5.2 Remove Rate Area . . . 159
8.5.3 Query Rate Area . . . 160
8.6 Class of Service . . . 161
8.6.1 Create Class of Service . . . 161
8.6.2 Delete Class of Service . . . 162
8.6.3 Modify Class of Service . . . 162
8.6.4 Query Class of Service . . . 163
8.7 Routing . . . 163
8.7.1 Create Routing . . . 164
8.7.2 Prioritize Routing . . . 165
8.7.3 Modify Routing . . . 167
8.7.4 Query Routing . . . 168
8.7.5 Unassign Routing . . . 170
8.8 Media Server . . . 172
8.8.1 Create Media Server . . . 172
8.8.2 Modify Media Server . . . 174
8.8.3 Remove Media Server . . . 176
8.8.4 Block / Unblock Media Server . . . 176

8.8.5 Query Media Server . . . 177
8.9 Media Server Circuits . . . 178
8.9.1 Create Media Server Circuits . . . 178
8.9.2 Remove Media Server Circuits . . . 179
8.9.3 Query Media Server Circuits . . . 180
8.10 CAC Group . . . 181
8.10.1 Create CAC Group . . . 181
8.10.2 Delete CAC Group . . . 185
8.10.3 Modify CAC Group . . . 185
8.10.4 Query CAC Group . . . 188
8.10.5 Add CAC Group Address . . . 191
8.10.6 Remove CAC Group Addr . . . 191
8.10.7 Get Ancestor Tree For CAC Group . . . 192
8.11 CAC Policy . . . 193
8.11.1 Create CAC Policy . . . 193
8.11.2 Delete CAC Policy . . . 196
8.11.3 Modify CAC Policy . . . 197
8.11.4 Query CAC Policy . . . 200
8.11.5 Get Bandwidth Usage for CAC Policy . . . 202
8.11.6 Create Group To Group CAC Policy . . . 203
8.11.7 Delete Group To Group CAC Policy . . . 206
8.11.8 Modify Group To Group CAC Policy . . . 206
8.11.9 Query Group To Group CAC Policy . . . 209
8.11.10 Get Bandwidth Usage for Group to Group CAC Policy . . . 211
8.11.11 Query CAC Monitoring . . . 212
8.12 Time Destination . . . 213
8.12.1 Create Time Destination . . . 213
8.12.2 Remove Time Destination . . . 215
8.12.3 Query Time Destination . . . 215
8.13 Day Schedule . . . 216
8.13.1 Create Day Schedule . . . 216
8.13.2 Assign Day Schedule . . . 217
8.13.3 Unassign Day Schedule . . . 218
8.13.4 Modify Day Schedule . . . 219
8.13.5 Remove Day Schedule . . . 220
8.13.6 Query Day Schedule . . . 221
8.14 Period Schedule . . . 222
8.14.1 Create Period Schedule . . . 222
8.14.2 Remove Period Schedule . . . 223
8.14.3 Modify Period Schedule . . . 223
8.14.4 Query Period Schedule . . . 224
8.15 Weekly Schedule . . . 225
8.15.1 Query Weekly Schedule . . . 225
8.16 Intercept . . . 227
8.16.1 Create Intercept . . . 227
8.16.2 Remove Intercept . . . 231
8.16.3 Modify Intercept . . . 232
8.16.4 Query Intercept . . . 233
8.17 Treatment . . . 235
8.17.1 Assign Treatment . . . 235
8.17.2 Modify Treatment . . . 240
8.17.3 Unassign Treatment . . . 242

8.17.4 Query Treatment . . . 244
8.17.5 Prioritize Treatment . . . 248
8.18 SIP Trusted Entity Ports . . . 249
8.18.1 Add SIP Trusted Entity Ports . . . 249
8.18.2 Remove SIP Trusted Entity Ports . . . 251
8.18.3 Get SIP Trusted Entity Ports List . . . 252
8.19 System SIP Entry . . . 254
8.19.1 Add System SIP Entry . . . 254
8.19.2 Modify System SIP Entry . . . 255
8.19.3 Delete System SIP Entry . . . 257
8.19.4 Get System SIP Entry Info . . . 258
8.19.5 Get System SIP Entry List . . . 260
8.20 Stat Counter . . . 260
8.20.1 Get Stat Counter Info . . . 260
8.20.2 Get Stat Counter List . . . 261
8.21 Database Job . . . 262
8.21.1 Create Database Job . . . 262
8.21.2 Cancel Database Job . . . 262
8.21.3 Delete Database Job . . . 263
8.21.4 Get Database Job List . . . 263
8.22 Calling Location . . . 265
8.22.1 Create Calling Location . . . 265
8.22.2 Modify Calling Location . . . 266
8.22.3 Remove Calling Location . . . 266
8.22.4 Query Calling Location . . . 267
8.23 Origin Destination . . . 268
8.23.1 Create Origin Destination . . . 268
8.23.2 Modify Origin Destination . . . 268
8.23.3 Remove Origin Destination . . . 269
8.23.4 Query Origin Destination . . . 269
8.24 Origin Route . . . 270
8.24.1 Create Origin Route . . . 270
8.24.2 Modify Origin Route . . . 271
8.24.3 Remove Origin Route . . . 271
8.24.4 Query Origin Route . . . 272
8.25 Get Generic Object Attributes . . . 273
8.26 Generic Query . . . 275
8.26.1 Generic Query Enhanced . . . 279
8.27 Code Process . . . 279
8.27.1 Create Code Process . . . 280
8.27.2 Modify Code Process . . . 282
8.27.3 Remove Code Process . . . 284
8.27.4 Query Code Process . . . 284
8.27.5 List Code Process . . . 285
8.28 ENUM Provisioning . . . 286
8.28.1 Create Enum Server . . . 286
8.28.2 Modify Enum Server . . . 287
8.28.3 Remove Enum Server . . . 288
8.28.4 Query Enum Server . . . 288
8.28.5 List Enum Server . . . 289
8.28.6 Create Operator Enum . . . 290
8.28.7 Modify Operator Enum . . . 290

8.28.8 Remove Operator Enum . . . 291
8.28.9 Query Operator Enum . . . 292
8.28.10 List Operator Enum . . . 292
8.29 File System Backup . . . 293
8.29.1 Create File System Backup . . . 293
8.29.2 Cancel File System Backup . . . 295
8.29.3 Query File System Backup . . . 296
8.30 Get DB Version . . . 298
8.30.1 Get DB Version Info . . . 298
8.31 Endpoint Location . . . 298
8.31.1 Create Endpoint Location . . . 299
8.31.2 Modify Endpoint Location . . . 299
8.31.3 Query Endpoint Location . . . 299
8.31.4 Remove Endpoint Location . . . 300
8.32 Endpoint Provider . . . 300
8.32.1 Create Endpoint Provider . . . 300
8.32.2 Modify Endpoint Provider . . . 301
8.32.3 Query Endpoint Provider . . . 301
8.32.4 Remove Endpoint Provider . . . 302

9 Subscriber . . . 303
9.1 Create Subscriber . . . 303
9.2 Get Subscriber Info . . . 320
9.2.1 Get Subscriber Info By Option . . . 333
9.2.2 Get Sub Info By Rel . . . 333
9.2.3 Get Subscriber RG Request . . . 334
9.3 Get Subscriber List . . . 336
9.4 Get TombStone Subscriber List . . . 343
9.5 Get Keyset Primary List . . . 344
9.6 Delete Subscriber . . . 345
9.7 Update Subscriber Status . . . 346
9.8 Update Subscriber Account Mgt Info . . . 347
9.9 Update Subscriber BG Info . . . 350
9.10 Update Subscriber Account User Info . . . 352
9.11 Update Subscriber Capabilities . . . 354
9.12 Update Keyset Info . . . 356
9.13 Update Subscriber Dn . . . 359
9.14 Update Subscriber Features . . . 361
9.15 Get Sub Tran Status . . . 364
9.16 Update Connection Info . . . 365
9.17 Update Subscriber Feature Profile . . . 366
9.18 Add Contact List . . . 367
9.19 Delete Contact List . . . 368
9.20 Get Contact List . . . 369
9.21 Update Mobile Info . . . 370

10 Features . . . 374
10.1 Account Code (CfAcctCode) . . . 374
10.2 Anonymous Caller Rejection (CfACR) . . . 375
10.3 Attendant Answering Position (CfBgNBCPU) . . . 375
10.4 Authorization Code (CfAuthCode) . . . 377
10.5 Call Transfer for Endpoint (SIP Applications) (CfEpCT) . . . 377
10.6 Completion of Calls on Busy Subscriber (CfCCBS) . . . 378

10.7 Completion of Calls on No Reply (CfCCNR) . . . 378
10.8 Call Forward Invalid (CfCFID) . . . 379
10.9 Call Forward System Internal External (CfCFSIE) . . . 379
10.10 Call Forward Unreachable (CfCFD) . . . 382
10.11 Call Forwarding - All Calls (CfCFV) . . . 382
10.12 Call Forwarding Busy Line (CfCFB) . . . 384
10.13 Call Forwarding Don't Answer (CfCFDA) . . . 386
10.14 Call Forwarding Voice Mail (CfVM) . . . 388
10.15 Call Pickup (CfBgCPU) . . . 390
10.16 Call Pickup Directed (CfCPUD) . . . 390
10.17 Called Party Name Delivery (CfCISNAME) . . . 391
10.18 Called Party Number Delivery (CfCISNUM) . . . 391
10.19 Calling Identity Delivery and Suppression (CfCIDS) . . . 391
10.20 Calling Name Delivery (CfCNAM) . . . 391
10.21 Calling Name Delivery (BG) (CfBgCNAM) . . . 391
10.22 Calling Name Delivery Blocking (CfCNAB) . . . 391
10.23 Calling Number Delivery (CfCND) . . . 392
10.24 Calling Number Delivery Blocking (CfCNDB) . . . 392
10.25 CSTA for ComAssistant (CfCSTA) . . . 392
10.26 Customer Originated Trace (CfMCT) . . . 393
10.27 Deny Terminating Feature (CfDTF) . . . 393
10.28 E9-1-1 LIN Administration (CfBgE911LIN) . . . 394
10.29 Executive Override (CfEXO) . . . 395
10.30 Distinctive Ringing (CfBgDR) . . . 396
10.31 Do Not Disturb (CfDND) . . . 397
10.32 Hot Desking (CfHotDesk) . . . 398
10.32.1 Hot Desking (BG) (CfBgHotDesk) . . . 399
10.33 Keyset Operation (CfBgKS) . . . 399
10.34 Station Controlled Conference (CfBgLC or CfSCC) . . . 400
10.35 Music On Hold (CfMOH) . . . 401
10.36 One Number Service (CfONS) . . . 402
10.37 Remote Activation of Call Forwarding (CfRACF) . . . 403
10.38 Remote Call Forwarding (CfRCF) . . . 404
10.39 Selective Call Acceptance (CfSCA) . . . 405
10.40 Selective Call Forwarding (CfSCF) . . . 406
10.41 Selective Call Rejecting (CfSCR) . . . 407
10.42 Serial Ringing (CfSERRNG) . . . 408
10.43 Simultaneous Ringing (CfSRS) . . . 409
10.44 Station Restrictions (Business Group) (CfSR) . . . 410
10.45 Station Speed Calling (CfSPCALL) . . . 411
10.46 System Speed Dial (CfBgSpDial) . . . 412
10.47 Time-of-Day Call Forwarding (CfECF) . . . 413
10.48 Toll and Call Restrictions (CfTRS) . . . 414
10.49 Traffic Measurement (CfBgTM) . . . 417
10.50 BG Silent Monitoring (CfBgSILM) . . . 417
10.51 Silent Monitoring Protection Feature (CfSILMProtection) . . . 418
10.52 Silent Monitoring Feature (CfSILM) . . . 418
10.53 Silent Monitoring Tag Feature (CfSILMTag) . . . 418
10.54 Call Forwarding Restrictions (CfCFR) . . . 419
10.55 Call Park to Server (CfCPS) . . . 420
10.56 Automatic Collect Call Blocking for Brazil . . . 420
10.57 Openscape Mobile Service (OSMO) . . . 421

10.58 Routing Service . . . 423
10.59 Network Wide Call Park to Server (Endpoint Profiles) . . . 424
10.60 CSTA Application external device name provider (Endpoint Profiles) . . . 425
10.61 Silent Monitoring Supervisor Feature (CfSILMS) . . . 426
10.62 Silent Monitoring Agent Feature (CfSILMA) . . . 427

11 Business Groups . . . 429
11.1 Create BG . . . 429
11.2 Update BG Status . . . 430
11.3 Add BG Main Number . . . 431
11.4 Delete BG Main Number . . . 432
11.5 Update BG Main Number . . . 433
11.6 Update BG Features . . . 434
11.7 Get BG Info . . . 436
11.7.1 Get BG Info By Options . . . 436
11.8 Get BG List . . . 437
11.9 Delete BG . . . 438
11.10 Get BG Attendant Numbers . . . 440
11.11 Delete BG Cpu . . . 440
11.12 Update BG Parms . . . 442
11.13 Business Group Departments . . . 443
11.13.1 Create BG Dept . . . 444
11.13.2 Delete BG Dept . . . 445
11.13.3 Modify BG Dept . . . 446
11.13.4 Get BG Dept List . . . 446
11.14 Business Group Subnets . . . 447
11.14.1 Get Bg Subnet Info . . . 448
11.14.2 Create Bg Subnet . . . 452
11.14.3 Get Bg Subnet List . . . 454
11.14.4 Delete Bg Subnet . . . 457
11.14.5 Update Bg Subnet . . . 458
11.14.6 Get Bg Subnet And LIN List . . . 462
11.15 Business Group Authorization Codes . . . 465
11.15.1 Create Auth Code . . . 465
11.15.2 Delete Auth Code . . . 466
11.15.3 Get Auth Code List . . . 467
11.15.3.1 Get Auth Code . . . 468
11.16 Modify Auth Code . . . 469
11.17 Private Numbering Plan . . . 469
11.18 Business Group System Speed Dial Lists . . . 470
11.18.1 Create Bg Speed Dial List . . . 471
11.18.2 Update Bg Speed Dial List . . . 472
11.18.3 Update Bg Speed Dial List Entries . . . 473
11.18.4 Delete Bg Speed Dial List Entries . . . 475
11.18.5 Delete Bg Speed Dial List . . . 476
11.18.6 Get Bg Speed Dial List . . . 477
11.18.7 Get Bg Speed Dial List Ids . . . 479
11.19 Business Group Parking Lots . . . 481
11.19.1 Create Parking Lot . . . 481
11.19.2 Update Parking Lot . . . 482
11.19.3 Delete Parking Lot . . . 483
11.19.4 Query Parking Lot Information . . . 484

11.19.5 Query Parking Lot List . . . 485
11.19.6 Query Parking Lot Usage List . . . 486
11.20 Multiple Media Servers per Branch Office . . . 488
11.20.1 Assign Media Server to Branch Office . . . 489
11.20.2 Unassign Media Server from Branch Office . . . 490
11.20.3 Get Media Server Assignment List . . . 491
11.21 Branch Office . . . 492
11.21.1 Create Branch Office . . . 493
11.21.2 Modify Branch Office . . . 494
11.21.3 Query Branch Office . . . 496
11.22 Remove Branch Office . . . 496
11.23 Remote Call Pickup Groups . . . 497
11.23.1 Add CPU Group . . . 497
11.23.2 Modify CPU Group . . . 498
11.23.3 Add Local CPU Members . . . 498
11.23.4 Modify Local CPU Members . . . 499
11.23.5 Remove Local CPU Members . . . 499
11.23.6 Add Remote Call Pickup Groups . . . 500
11.23.7 Modify Remote Call Pickup Groups . . . 501
11.23.8 Remove Remote Call Pickup Groups . . . 501
11.23.9 Get Call Pickup Group List . . . 502
11.23.10 Get Call Pickup Group Member List . . . 503
11.23.11 Get Local CPU Members List . . . 504
11.23.12 Get Remote CPU Members List . . . 504
11.24 Community Groups . . . 505
11.24.1 Add Community Group . . . 506
11.24.2 Modify Community Group . . . 506
11.24.3 Remove Community Group . . . 507
11.24.4 Query Community Group . . . 507
11.24.5 Add Community Group Member . . . 508
11.24.6 Modify Community Group Member . . . 509
11.24.7 Remove Community Group Member . . . 509
11.24.8 Query Community Group Member . . . 510
11.25 BG Suite . . . 511
11.25.1 Create BG Suite . . . 512
11.25.2 Configuration of the Extension Dialing . . . 513
11.25.3 Prefix Access Codes . . . 513
11.25.4 Destination Codes for Extension dialing . . . 514
11.25.5 Extensions . . . 514

12 Numbering Plans . . . 516
12.1 Numbering Plan . . . 516
12.1.1 Create Number Plan . . . 516
12.1.2 Update Number Plan . . . 517
12.1.3 Delete Number Plan . . . 518
12.1.4 Get Number Plan List . . . 519
12.1.5 Get Number Plan ID Info . . . 521
12.1.6 Get Extended Numbering Plan . . . 521
12.2 Destination Codes . . . 523
12.2.1 Create Dest Code . . . 523
12.2.2 Modify Dest Code Info . . . 527
12.2.3 Delete Dest Code . . . 530

12.2.4 Get Dest Code List . . . 532
12.2.5 Get Dest Code . . . 534
12.2.6 Get Dest Service List . . . 534
12.3 Prefix Access Code . . . 535
12.3.1 Create Prefix Access Code . . . 535
12.3.2 Modify Prefix Access Code . . . 540
12.3.3 Delete Prefix Access Code . . . 545
12.3.4 Get Prefix Access Code List . . . 546
12.3.5 Get Prefix Access Code One . . . 549
12.4 Private Numbering Plan Location Codes . . . 550
12.4.1 Create PNP Location Code . . . 550
12.4.2 Modify PNP Location Code . . . 551
12.4.3 Delete PNP Location Code . . . 552
12.4.4 Get PNP Location Code List . . . 553
12.5 PNP Extension . . . 554
12.5.1 Create PNP Extension . . . 554
12.5.2 Modify PNP Extension . . . 556
12.5.3 Delete PNP Extension . . . 557
12.5.4 Get PNP Extension List . . . 558

13 Number Modification . . . 560
13.1 DNM Prefixes . . . 560
13.1.1 Create DNM Prefix . . . 560
13.1.2 Modify DNM Prefix . . . 562
13.1.3 Delete DNM Prefix . . . 563
13.1.4 Query DNM Prefix . . . 565
13.1.5 Query DNM PrefixList . . . 566
13.2 DNM Definitions . . . 568
13.2.1 Create DNM Definition . . . 569
13.2.2 Modify DNM Definition . . . 571
13.2.3 Delete DNM Definition . . . 572
13.2.4 Query DNM Definition . . . 574
13.2.5 Query DNM Definition . . . 575
13.3 DNM Modification . . . 577
13.3.1 Create DNM Modification . . . 579
13.3.2 Modify DNM Modification . . . 582
13.3.3 Delete DNM Modification . . . 583
13.3.4 Query DNM Modification . . . 584
13.4 DNM Normalizations . . . 586
13.4.1 Input Patterns . . . 587
13.4.2 Output Expression . . . 588
13.4.3 Create DNM Normalization . . . 589
13.4.4 Modify DNM Normalization . . . 591
13.4.5 Delete DNM Normalization . . . 593
13.4.6 Query DNM Normalization . . . 594
13.5 DNM Conversions . . . 596
13.5.1 Create DNM Conversion . . . 597
13.5.2 Modify DNM Conversion . . . 599
13.5.3 Delete DNM Conversion . . . 601
13.5.4 Query DNM Conversion . . . 602
13.6 DNM Local Toll . . . 604
13.6.1 Create DNM LocalToll . . . 605

13.6.2 Modify DNM LocalToll . . . 607
13.6.3 Delete DNM LocalToll . . . 609
13.6.4 Query DNM LocalToll . . . 610

14 Feature Profiles . . . 613
14.1 Create Feature Profile . . . 613
14.2 Update Feature Profile . . . 614
14.3 Delete Feature Profile . . . 615
14.4 Get Feature Profile List . . . 616
14.5 Get Feature Profile . . . 617

15 Hunt Groups . . . 619
15.1 Terminals . . . 620
15.2 Hunt Group Phone Services . . . 621
15.2.1 Create Mlhg . . . 622
15.2.2 Modify Mlhg Info . . . 626
15.2.3 Delete Mlhg . . . 632
15.2.4 Get Mlhg Info . . . 632
15.2.5 Add Sub To Mlhg . . . 637
15.2.6 Create Mlhg Term . . . 640
15.2.7 Modify Mlhg TermInfo . . . 640
15.2.8 Delete Mlhg Term . . . 642
15.2.9 Get Mlhg TM Data . . . 643

16 Endpoint Profiles . . . 644
16.1 Create Ep Profile . . . 644
16.1.1 Update Ep Mgt Info . . . 648
16.1.2 Update Ep Capabilities . . . 651
16.1.3 Update Ep Features . . . 653
16.1.4 Get Ep Info . . . 653
16.1.5 Get Ep Info By Release . . . 655
16.1.6 Get EP List . . . 655
16.1.7 Delete Ep Profile . . . 657

17 Background Job Processing Interfaces . . . 658
17.1 Jobs . . . 658
17.1.1 Get Job Info . . . 658
17.1.2 Get Job List . . . 660
17.1.3 Get Job List By Option . . . 660
17.1.4 Cancel Job . . . 661
17.1.5 Generic Job . . . 662

18 Dynamic CAC Router Management . . . 664
18.1 Notify CAC Primary Link Status . . . 664
18.2 Get CAC Primary Link Status . . . 664
18.3 Get Provisioned CAC Primary Links . . . 665
18.4 Reset All CAC Primary Links . . . 666

19 Executive Assistant Functionality with Cockpit Application . . . 668
19.1 Executive Assistant Groups . . . 668
19.1.1 Create Executive Assistant Group . . . 668
19.1.2 Delete Executive Assistant Group . . . 669
19.1.3 Update Executive Assistant Group . . . 670
19.1.4 Get List of Executive Assistant Groups . . . 672
19.1.5 Get Extended Executive Assistant Group List . . . 673

19.2 Executive Assistant Group Members . . . 674
19.2.1 Create Executive Assistant Group Member . . . 674
19.3 Delete Executive Assistant Group Member . . . 675
19.3.1 Get List of Executive Assistant Group Members . . . 676
19.4 Executive Assistant Group Member Relations . . . 677
19.4.1 Create Executive Assistant Group Member Relation . . . 677
19.4.2 Delete Executive Assistant Group Member Relation . . . 678
19.4.3 Get Executive Assistant Group Member Relation List . . . 679
19.5 Executive Assistant Group Member Auxiliary Lines . . . 680
19.5.1 Create Executive Assistant Group Member Auxiliary Line . . . 681
19.5.2 Delete Executive Assistant Group Member Auxiliary Line . . . 682
19.5.3 Get Executive Assistant Group Member Relations . . . 683
19.6 Executive Assistant Cockpit . . . 684
19.6.1 Get Executive Assistant Cockpit List . . . 684
19.6.2 Get Executive Assistant Cockpit Configuration Data . . . 685
19.6.3 Get Executive Assistant Cockpit Dynamic Data . . . 686
19.6.4 Update Executive Assistant Cockpit Dynamic Data . . . 687

20 Upgrade and Remote Patching . . . 690
20.1 Get Node Info . . . 690
20.2 Get Node Patch Set List . . . 691
20.3 Get SIS Patch Set List . . . 692
20.4 Upgrade . . . 693
20.5 Upgrade Second Node . . . 695
20.6 Get Upgrade Status . . . 695
20.7 Upgrade Commit . . . 697
20.8 Upgrade Fallback . . . 697
20.9 Get Node Health . . . 698
20.10 Query Operation Mode . . . 699
20.11 Modify Operation Mode . . . 700
20.12 Get Database Role . . . 701

21 Traffic Types and Classes of Restrictions . . . 703
21.1 Create Traffic Type . . . 703
21.2 Modify Traffic Type . . . 705
21.3 Delete Traffic Type . . . 706
21.4 Query Traffic Type . . . 707
21.5 Create Class Of Restrictions . . . 708
21.6 Modify Class Of Restrictions . . . 709
21.7 Delete Class Of Restrictions . . . 710
21.8 Query Class Of Restrictions . . . 710
21.9 Query Class Of Restrictions Info List . . . 711
21.10 Query COR Usage List . . . 712
21.11 Query COR Group Key List . . . 713
21.12 CSTA Applications . . . 714
21.12.1 GetCSTAApplicationList . . . 715
21.12.2 CreateCSTAApplication . . . 717
21.12.3 ModifyCSTAApplication . . . 719
21.12.4 QueryCSTAApplication . . . 722
21.12.5 GetCSTAActiveDeviceList . . . 725
21.12.6 DeleteCSTAApplication . . . 727
21.12.7 GetCSTASMInfo . . . 728

22 CSTA Applications . . . 731
22.1 Get CSTA Application List . . . 731
22.2 Create CSTA Application . . . 733
22.3 Modify CSTA Application . . . 738
22.4 Query CSTA Application . . . 741
22.5 Get CSTA Active Device List . . . 745
22.6 Delete CSTA Application . . . 747
22.7 Get CSTASM Info . . . 748

23 Alarms and Log Events . . . 750
23.1 Alarms . . . 750
23.1.1 SOAP Server Process Exit . . . 750
23.1.2 SOAP Server Process Startup . . . 751
23.1.3 Submgt Schedule Process Exit . . . 751
23.1.4 Submgt Schedule Process Startup . . . 752
23.1.5 Error Removing Old Jobs . . . 753
23.1.6 Old Jobs Removed Successfully . . . 753
23.2 Log Events . . . 754
23.2.1 Error Accessing the Database . . . 755
23.2.2 Provisioning Error . . . 755
23.3 Display of OSV Error Provisoning Logs . . . 755

24 Import and Export of SOAP Provisioned Data . . . 758
24.1 Export and Mass Provisioning . . . 758
24.1.1 Importing Subscriber Data . . . 758
24.1.2 Exporting Subscriber Data . . . 760
24.2 Supported Commands . . . 764
24.2.1 Add Features . . . 765
24.2.1.1 Supported Features . . . 765
24.2.2 Remove Features . . . 788
24.2.3 Modify Features . . . 789
24.2.4 Deny Features . . . 789
24.2.5 Add Feature Profile Features . . . 789
24.2.6 Remove Feature Profile Features . . . 790
24.2.7 Modify Feature Profile Features . . . 790
24.2.8 Add Business Group Features . . . 790
24.2.8.1 Supported Features . . . 791
24.2.9 Remove Business Group Features . . . 792
24.2.10 Add End Point Profile Features . . . 792
24.2.10.1 Supported Features . . . 792
24.2.11 Create CSTA Application . . . 798
24.2.12 Modify CSTA Application . . . 798
24.2.13 Delete CSTA Application . . . 799
24.2.14 Delete End Point Profile Features . . . 799
24.2.15 Create Subscriber . . . 799
24.2.16 Delete Subscriber . . . 804
24.2.17 Update Subscriber Directory Number . . . 805
24.2.18 Update Subscriber Capabilities . . . 805
24.2.19 Create Hunt Group . . . 805
24.2.20 Modify Mlhg . . . 806
24.2.21 Remove Hunt Group . . . 806
24.2.22 Add Subscriber To Hunt Group . . . 807
24.2.23 Modify Hunt Group Terminal . . . 807

24.2.24 Remove Hunt Group Terminal . . . 807
24.2.25 Create Feature Profile . . . 807
24.2.26 Delete Feature Profile . . . 808
24.2.27 Update Subscriber Feature Profile. . . . 808
24.2.28 Update Subscriber Account Management Info . . . 808
24.2.29 Update Account Business Group Information . . . 808
24.2.30 Update Subscriber PIC Information . . . 809
24.2.31 Update Subscriber QoS Profile Information. . . . 809
24.2.32 Update Subscriber Status . . . 810
24.2.33 Update Subscriber User information . . . 810
24.2.34 Update Connection Information for a SIP Subscriber . . . 810
24.2.35 Add Contact Information . . . 811
24.2.36 Remove Contact Information . . . 813
24.2.37 Create Business Group . . . 815
24.2.38 Delete Business Group . . . 816
24.2.39 Update Business Group Status . . . 816
24.2.40 Add Business Group Main Number . . . 816
24.2.41 Delete Business Group Main Number . . . 816
24.2.42 Update Business Group MDR Information. . . . 817
24.2.43 Update Business Group Media Server Information . . . 817
24.2.44 Update Subscriber Keyset Info . . . 817
24.2.45 Create Business Group Department . . . 819
24.2.46 Delete Business Group Department . . . 819
24.2.47 Create Business Group E9-1-1 Subnet . . . 819
24.2.48 Delete Business Group E9-1-1 Subnet . . . 820
24.2.49 Update Business Group E9-1-1 Subnet. . . . 820
24.2.50 Create Endpoint Profile . . . 821
24.2.51 Delete Endpoint Profile . . . 821
24.2.52 Update Endpoint Profile Management Information . . . 822
24.2.53 Update Endpoint Profile PICs . . . 822
24.2.54 Update Endpoint Profile QoS Information . . . 823
24.2.55 Create QoS Profile. . . . 823
24.2.56 Modify QoS Profile. . . . 824
24.2.57 Delete QoS Profile . . . 825
24.2.58 Create Destination Code . . . 825
24.2.59 Modify Destination Code . . . 826
24.2.60 Remove Destination Code . . . 826
24.2.61 Create Number Plan . . . 826
24.2.62 Remove Number Plan . . . 827
24.2.63 Create Authorization Code. . . . 827
24.2.64 Delete Authorization Code . . . 827
24.2.65 Create Prefix Access Code . . . 828
24.2.66 Create PNP Location Code . . . 828
24.2.67 Update PNP Location Code. . . . 829
24.2.68 Remove PNP Location Code . . . 829
24.2.69 Update Prefix Access Code . . . 829
24.2.70 Remove Prefix Access Code . . . 829
24.2.71 Create PNP Extension . . . 830
24.2.72 Update PNP Extension . . . 830
24.2.73 Remove PNP Extension . . . 830
24.2.74 Create Display Number Definition . . . 830
24.2.75 Delete Display Number Definition . . . 831

24.2.76 Create Display Number Prefix . . . 831
24.2.77 Modify Display Number Prefix . . . 831
24.2.78 Delete Display Number Prefix. . . . 832
24.2.79 Create Display Number Modification . . . 832
24.2.80 Modify Display Number Modification . . . 832
24.2.81 Delete Display Number Modification. . . . 832
24.2.82 Update Business Group Common Numbering Plan . . . 833
24.2.83 Update Business Group Default QoS profile . . . 833
24.2.84 Create CAC Group . . . 833
24.2.85 Delete CAC Group . . . 834
24.2.86 Remove CAC Group Address. . . . 835
24.2.87 Add CAC Group Address . . . 836
24.2.88 Create CAC Policy . . . 837
24.2.89 Delete CAC Policy . . . 838
24.2.90 Create Group To Group CAC Policy. . . . 838
24.2.91 Delete Group To Group CAC Policy . . . 839
24.2.92 Create Business Group Speed Calling List . . . 839
24.2.93 Update Business Group Speed Calling List . . . 839
24.2.94 Delete Business Group Speed Calling List. . . . 839
24.2.95 Delete Business Group Speed Calling List Entries. . . . 840
24.2.96 Update Business Group Speed Calling List Entries . . . 840
24.2.97 Assign Day Schedule . . . 840
24.2.98 Assign Treatment . . . 840
24.2.99 Block Media Gateway . . . 841
24.2.100 Block Trunk. . . . 841
24.2.101 Block Trunk Group . . . 841
24.2.102 Create Alias . . . 842
24.2.103 Create Calling Location . . . 842
24.2.104 Create Class Of Service . . . 842
24.2.105 Create Code Process . . . 842
24.2.106 Create Day Schedule . . . 843
24.2.107 Create Destination . . . 843
24.2.108 Create Endpoint . . . 843
24.2.109 Create HomeDN . . . 845
24.2.110 Create Intercept . . . 845
24.2.111 Create Media Gateway Circuits . . . 845
24.2.112 Create Media Gateway . . . 846
24.2.113 Create Office Code . . . 846
24.2.114 Create Origin Destination . . . 847
24.2.115 Create Origin Route . . . 847
24.2.116 Create Period Schedule . . . 847
24.2.117 Create Rate Area . . . 847
24.2.118 Create Route . . . 848
24.2.119 Create Time Destination . . . 848
24.2.120 Create Trunk. . . . 848
24.2.121 Create Trunk Group . . . 849
24.2.122 Modify Class Of Service . . . 850
24.2.123 Modify Code Process . . . 850
24.2.124 Modify Day Schedule . . . 850
24.2.125 Modify Destination . . . 850
24.2.126 Modify Endpoint . . . 851
24.2.127 Modify HomeDN . . . 852

24.2.128 Modify Intercept . . . 852
24.2.129 Modify Media Gateway . . . 852
24.2.130 Modify Office Code . . . 853
24.2.131 Modify Origin Destination . . . 853
24.2.132 Modify Origin Route . . . 853
24.2.133 Modify Period Schedule . . . 854
24.2.134 Modify Route . . . 854
24.2.135 Modify Treatment . . . 855
24.2.136 Modify Trunk . . . 855
24.2.137 Modify Trunk Group . . . 855
24.2.138 Prioritize Route . . . 856
24.2.139 Prioritize Treatment . . . 856
24.2.140 Remove Alias . . . 857
24.2.141 Remove Calling Location . . . 857
24.2.142 Remove Class Of Service . . . 857
24.2.143 Remove Code Process . . . 857
24.2.144 Remove Day Schedule . . . 858
24.2.145 Remove Destination . . . 858
24.2.146 Remove Endpoint . . . 858
24.2.147 Remove HomeDN . . . 858
24.2.148 Remove Intercept . . . 859
24.2.149 Remove Media Gateway Circuits . . . 859
24.2.150 Remove Media Gateway . . . 859
24.2.151 Remove Office Code . . . 859
24.2.152 Remove Origin Destination . . . 859
24.2.153 Remove Origin Route . . . 860
24.2.154 Remove Period Schedule . . . 860
24.2.155 Remove Rate Area . . . 860
24.2.156 Remove Time Destination . . . 860
24.2.157 Remove Trunk . . . 861
24.2.158 Remove Trunk Group . . . 861
24.2.159 Unassign Day Schedule . . . 861
24.2.160 Unassign Route . . . 861
24.2.161 Unassign Treatment . . . 862
24.2.162 Create Enum Server . . . 862
24.2.163 Modify Enum Server . . . 862
24.2.164 Remove Enum Server . . . 863
24.2.165 Create Operator Enum . . . 863
24.2.166 Modify Operator Enum . . . 863
24.2.167 Remove Operator Enum . . . 863
24.2.168 Set RTP Parameters . . . 864
24.2.169 Create Disconnect DN . . . 864
24.2.170 Update Dn Reservation . . . 864
24.2.171 Create a Parking Lot . . . 865
24.2.172 Update a Parking Lot . . . 865
24.2.173 Delete a Parking Lot . . . 865
24.2.174 Add Community Group . . . 865
24.2.175 Add Community Group Member . . . 865
24.2.176 Modify Community Group . . . 866
24.2.177 Modify Community Group Member . . . 866
24.2.178 Remove Community Group Member . . . 866
24.2.179 Remove Community Group . . . 866

24.2.180 Update Subscriber Mobile Info . . . 867
24.2.181 Create Traffic Type . . . 867
24.2.182 Modify Traffic Type . . . 867
24.2.183 Delete Traffic Type . . . 867
24.2.184 Create Class Of Restrictions . . . 867
24.2.185 Modify Class Of Restrictions. . . . 870
24.2.186 Create Call Pickup Group . . . 872
24.2.187 Modify Call Pickup Group . . . 872
24.2.188 Delete Call Pickup Group . . . 872
24.2.189 Add Local Call Pickup Group member . . . 872
24.2.190 Modify Local Call Pickup Group member . . . 873
24.2.191 Remove local Call Pickup Group member . . . 873
24.2.192 Add Remote Call Pickup Group . . . 873
24.2.193 Modify Remote Call Pickup Group . . . 873
24.2.194 Remove Remote Call Pickup Group . . . 874
24.2.195 Create Branch Office . . . 874
24.2.196 Modify Branch Office . . . 875
24.2.197 Remove Branch Office . . . 877

25 Subscriber Provisioning Result Information . . . 878
25.1 Result Code Information . . . 878
25.2 ResultMsgNumber . . . 880
25.3 Result Text Strings. . . . 880
25.4 ResultTOY . . . 881
25.5 Result Codes/Result Text Settings . . . 881

26 Using SOAP to Provision the OpenScape Voice Assistant . . . 882
26.1 RTP Configuration Management . . . 882
26.1.1 Set an RTP Parameter . . . 885
26.1.2 Display the List of Accessible RTP Parameters . . . 885
26.2 Call Statistics . . . 890
26.2.1 Display the Counter Value . . . 891
26.2.2 Display the Counter Status . . . 892
26.2.3 Display the Counter Names . . . 892
26.3 Database Backup and Restore . . . 893
26.3.1 Create Database Job Request . . . 894
26.3.2 Cancel Database Job Request . . . 894
26.3.3 Get Database Job List . . . 894
26.3.4 Values for Job Request . . . 894
26.4 Managing Time Destinations Tables . . . 895
26.4.1 Time Destination. . . . 895
26.4.2 Day Schedule . . . 896
26.4.3 Period Schedule . . . 898
26.5 SIP Signaling Manager . . . 900
26.5.1 Assign Port Range . . . 900
26.5.2 Unassign Port Range . . . 901
26.5.3 Display Port Ranges . . . 901
26.6 Billing and CDR Generation . . . 902
26.7 Tones and Announcements . . . 902
26.7.1 Intercepts . . . 902
26.7.1.1 Create an Intercept . . . 902
26.7.1.2 Modify an Intercept . . . 903
26.7.1.3 Remove an Intercept . . . 903

26.7.1.4 Display an Intercept . . . 903
26.7.2 Treatments . . . 904
26.7.2.1 Assign a Treatment . . . 904
26.7.2.2 Modify an Existing Treatment . . . 905
26.7.2.3 Unassign a Treatment . . . 905
26.7.2.4 Display a Treatment . . . 906
26.8 PRIMECLUSTER (PCL) Management . . . 906

27 RTP Parameters Relevant to Subscriber Provisioning . . . 908

28 WSDL and Example SOAP Messages . . . 913
28.1 WSDL Schema Definition . . . 913
28.2 Interfacing to the System using the WSDL . . . 913
28.3 Sample Messages . . . 913
28.3.1 Get Version Sample Messages . . . 914
28.3.1.1 Get Version Request . . . 914
28.3.1.2 Get Version Response . . . 914
28.3.2 Create Subscriber Sample Messages . . . 915
28.3.2.1 Create Subscriber Request . . . 915
28.3.2.2 Create Subscriber Response . . . 918
28.3.3 Get Subscriber Info Sample Messages . . . 919
28.3.3.1 Get Subscriber Info Request . . . 919
28.3.3.2 Get Subscriber Info Response . . . 919
28.3.4 Update Subscriber Features Sample Messages . . . 923
28.3.4.1 Update Subscriber Features Request . . . 923
28.3.4.2 Update Subscriber Features Response . . . 924
28.3.5 Update Subscriber PIC Sample Messages . . . 925
28.3.5.1 Update Subscriber PIC Request . . . 925
28.3.5.2 Update Subscriber PIC Response . . . 926
28.3.6 Update Subscriber Account Management Info Sample Messages . . . 926
28.3.6.1 Update Subscriber Account Management Info Request . . . 926
28.3.6.2 Update Subscriber Account Management Info Response . . . 927
28.3.7 Update Subscriber Account User Info Sample Messages . . . 928
28.3.7.1 Update Subscriber Account User Info Request . . . 928
28.3.7.2 Update Subscriber Account User Info Response . . . 928
28.3.8 Update Subscriber Status Sample Messages . . . 929
28.3.8.1 Update Subscriber Status Request . . . 929
28.3.8.2 Update Subscriber Status Response . . . 929
28.3.9 Update Subscriber Connection Info Sample Messages . . . 930
28.3.9.1 Update Subscriber Connection Info Request . . . 930
28.3.9.2 Update Subscriber Connection Info Response . . . 931
28.3.10 Update Subscriber DN Sample Messages . . . 932
28.3.10.1 Update Subscriber DN Request . . . 932
28.3.10.2 Update Subscriber DN Response . . . 932
28.3.10.3 Delete Subscriber Request . . . 933
28.3.10.4 Delete Subscriber Response . . . 933
28.3.11 Get COS List Sample Messages . . . 934
28.3.11.1 Get COS List Request . . . 934
28.3.11.2 Get COS List Response . . . 934

29 FeatureList.dat 936

A Time Zones 938

B Test Tools 943
B.1 TestSendSoapReq 943
B.2 TestSoap.ksh (soapCli) 944

C Performance 945
C.1 Priority Processing and Congestion Controls 946

1 Introduction

Subscriber provisioning is the management of subscriber data on the switch, which is managed by a component referred to as the SOAP Server.

The SOAP (Simple Object Access Protocol) Server is an integrated component of the OpenScape Voice system whose function is to handle provisioning requests for subscriber, endpoint, and business group data, including the creation, deletion, modification, and display of these entities and their associated features, or services. The SOAP Server expects SOAP/XML requests conforming to the SOAP Server WSDL (Web Services Definition Language) file, which is included as a product deliverable and is made available to external interface partners. In particular, the OpenScape Voice Assistant uses this interface to transmit provisioning requests from the user to the OpenScape Voice system.

Subscriber provisioning is the management of subscriber data on the system which consists of account creation and management of subscriber and Business Group data. The provisioning interface handles this management and is intended to be middleware between the Customer Relationship Management (CRM) system and the call flow. This book does not provide descriptions of how the features are processed during call flow.

1.1 Deployment Network Configuration

The OpenScape Voice system is usually installed as a clustered pair of servers. Subscriber provisioning is available on both machines of the cluster. Furthermore, the provisioning interface supports multiple ports (n, n+1..n+m). It is recommended that operator and subscriber traffic use different ports. The operator traffic should use the virtual IP address for the cluster and port n, while subscribers use the same IP address and port n+1. In this manner, traffic will be distributed and operators will not be blocked by subscriber traffic. If there is an electronic interface (flow through provisioning), it should use the same connection information as the operator traffic.

1.2 Subscriber Interactions

Subscribers may interact because of their feature or service data. For example, a subscriber with Speed Dialing can reference another subscriber's Speed Dialing List. These relationships may cause certain service features or even the basic processing not to work correctly when the referenced subscriber changes his or her service or features.

In this case, the SOAP Server methods process the request without regard to other subscribers that might reference the changing subscriber. In other words, subscriber A might have a Call Forward feature, specifying subscriber B as the

forward-to destination. If subscriber B is deleted, disconnected, or otherwise modified such that it is no longer a valid forward-to destination, there is no notification of this fact, nor is there any mechanism by which subscriber A's data is audited or updated.

1.2.1 Service Interactions

Some services interact with each other: some may be assigned together, but interact during call processing such that the functionality of one overrides the other. In other cases, features may not be assigned together because they are mutually exclusive, and finally in some cases, features are dependent on other features. In the latter two cases, checks are performed during provisioning to block the assignment of features that are mutually exclusive, and to prevent the removal of features on which other features depend.

Although some particular feature interactions may be mentioned in this document, this document is not intended to contain a complete list of the interactions. The provisioning checks are designed to match the interactions specified in the *OpenScape Voice Feature Description Guide*; therefore, refer to that document for further information about feature compatibility and dependencies.

1.3 Subscriber Provisioning Network Architecture Layout

The subscriber profile data is split between the CRM system and the OpenScape Voice system. The CRM is the "owner" of the complete subscriber record while the OpenScape Voice system maintains only a run-time subset of that data necessary to process calls. That information which is not run time would not be sent to the OpenScape Voice system by the CRM. The CRM would query the OpenScape Voice system whenever it needed specific runtime data.

For example, consider the subscriber record containing the following information:

- Subscriber Alias (DN)
- Subscriber Address
- Subscriber Call Forwarding Variable Number

The CRM would provision the line to the OpenScape Voice system and would need to send the DN and, optionally, an initial Call Forwarding Variable number. It would not send the address but would maintain that address in the CRM itself. The Call Forwarding Variable Number, though, would be maintained on the OpenScape Voice system. Whenever the CRM needs that data value, it would request it from the OpenScape Voice system.

The idea is that CRM systems are currently existing (CEON, MetaSolv, Convergys, Liberate, etc. are some vendors of these systems) while others, have internally built systems. Electronic integration with those existing systems is desirable to avoid requiring operators to double key into the system-provided provisioning screens.

1.3.1 Small Medium Enterprise

The Small Medium Enterprise (SME) is a framework that was integrated in soapServer to cover some special needs for the Enterprise environment. In the Enterprise environment, the OpenScape Voice system uses one web client, the OpenScape Voice Assistant, to do both subscriber provisioning and system administration. With the introduction of the SME framework, and in conjunction with the OpenScape Voice Assistant, the soapServer now supports (and will continue to add additional support functionality for) system administration.

1.3.2 OpenScape Voice Assistant

OpenScape Voice Assistant lets you administer the entire OpenScape Voice communication system. It is a Web-based application accessed via a browser. It can administer a OpenScape Voice communication system with 300 to 5,000 users when it is installed in the same Linux machine as the OpenScape Voice system (internal) and a system with up to the maximum number of users supported from the OpenScape Voice system when is installed in a Linux machine different from the OpenScape Voice system (external).

1.3.3 SIP Network Servers

The OpenScape Voice system supports SIP Network Servers. These network entities are much like PBXs. They are defined on the OpenScape Voice system using the Assistant, and profiles and feature profiles are created using the Assistant or the SOAP interface.

These SIP Network Server profiles are called Endpoint Profiles.

2 Changes Between Version V6 and V7

The following are lists of changes that have been made in the SOAP interface from Version 6 and Version 7. Note that the expected use of this information is to determine if an existing SOAP client should be updated to take advantage of the new functionality. Once it is determined from this high-level summary that a particular change should be addressed in a SOAP client, the old and new WSDL files should be compared to determine the exact interface changes that have occurred. Not all changes listed here apply to all WSDL variants. For example, changes related to Business Groups would not affect the Subscriber Self-Care WSDL; updates to subscriber features would not affect the Network Management WSDL,etc.

Removals:

- ServiceFeatureType Pbx and PbxFeatureProfile.
- ServiceIdType =PbxParent, PbxChild, Pbxrange, PbxDNRangeInfo
- GetPbxParentInfo all values, UpdatePbxParentConnectionInfo, CreatePbxInfo, isPbxParent, UpdatePbxParentInfo
- TranslationResult = XLARC-PIC1Required, XLARC-PIC2Required, XLARC-PICRequired.
- DestType = XLADT-GKAlias, XLADT-LNP, XLADT-AINTrgg.
- DestCodeTriggersType all values,
- LNPCode all values
- IPConnRegTypes = StaticDynamic
- FeatureProfileType = FeatureProfilePbx
- PbxChildType all values
- DNFilter
  - DnPortedIn
  - DnNoPortedIn
  - DnPortedOut"
  - DnNoPortedOut
  - DnNonPorted"
  - DnNoNonPorted
  - DnImported
  - DnNotImported

  - DnNative
  - DnNoNative
- LnpStatus
- TransFlags all values
- IPPhoneAttribute = Reserved1
- UpdateSubscriberPortReq,UpdateSubscriberPortRequest, UpdateSubscriberPortResult,
- DisconnectSubscriberRequest, DisconnectSubscriberResult
- UpdateSubscriberPort2Request, DisconnectSubscriber2Request
- DeleteBGCpu2Request
- HGFilterType=HGFilterByBOName
- IPPhoneAttribute=AcceptBillingNumber, Reserved2, Reserved3
- Removed obsolete info for classic import/export format

Additions:

- CRSRoutingStatus
- SubscriberLicenseType = OSMOSubscriber, SLESLicense
- PilotDnType
- Feature = CfSILMS, CfSILMA
- GetSubscriberListDataOption = SubListDataInclIsMainDevice
- LicenseType = AllLicense, OSMOLicense
- LicenseUsageData = VoiceAllData, VocieOSMOData
- HGFilterType=HGFilterbyBOName
- GetBgSubnetListDataOption
- CSTAApplicationType, CSTAApplicationSessionStatus, CSTAApplicationConnectionStatus, CSTADeviceFilter, CSTADeviceSortType, CSTADeviceType, CSTAApplicationSortType,CSTASMStatus,
- CSTAApplicationIdentification, CSTAApplicationSOAPHeartBeat, CSTAApplicationOperationalData, CreateCSTAApplicationResult,
- CORGroupKeyList, QueryCORGroupKeyListRequest, QueryCORGroupKeyListResult,

- CreateCSTAApplicationRequest, ModifyCSTAApplicationRequest, ModifyCSTAApplicationResult, DeleteCSTAApplicationRequest, DeleteCSTAApplicationResult, CSTAApplicationStatus
- MonitoringSessionsData
- ToneMonitoringData
- CfSILMS-locPres
- CfSILMA, CfSILMS
- CfCRS RoutingStatus
- ListOfFeatures CfSILMS, CfSILMA
- IPPhoneAttribute = Reserved4, DisableLongCallAudit, Reserved5, Reserved6
- MlhgInfo PilotInfor = PilotDnType, NoInterceptAnnouncement, AllowCallsToOverflow, AllowUnansweredCallsToRequeue
- FeatureInfo = CCBSAssigned, CCNRAssigned
- GetSubscriberListEntry = MainDevice
- BOName in CreateMlhgRequest, ModifyMlhgInfoRequest, AddSubToMlhgRequest2
- OSMOLicenseInfo
- HGMember
  - NoAnsAdvTime
  - OverFlowDN
  - PilotDnType
  - MlhgTermList
- OsbSyncDataKey, OsbSyncData, OsbSyncDataList, GetOsbSyncDataRequest, GetOsbSyncDataResult,
- CfBgSILM = MaxNumberOfSupervisors
- BgSubnetData = AssociatedEndpointId, RepresentativeEndpointName
- GetBgSubnetListDataOptionListEntry, GetBgSubnetListDataExtended
- GetBgSubnetAndLinListRequest, BgSubnetAndLinListData, GetBgSubnetAndLinListData, GetBgSubnetAndLinListResult
- CfCNIM
- EndpointAttributes = UseEndpointDefaultHomeDnAsAuthNumber, UseSubscriberHomeDnAsAuthNumber, CSTADeviceId

- Endpoint element= ManagementIpAddress, CSTADeviceId
- ImpactLevel
- TransportProtocol
- PFR operations (Packet Filter Rules)
- CRSRoutingStatus
- HuntType Parallel
- GetSubscriberListDataOption
  - SubListDataInclIsMainDevice
  - SubListDataInclCpuName
  - SubListDataInclMlhgList
- CpuGroupCIDPresentation
- CfSILMS attributes
- OvlMgrSysInfo Virtualized
- CfBgSILM MaxNumberOfSupervisors
- UpNewIpV6Address
  - UpdateBgSubnetRequest
  - NewIpV6PrefixLen
- CpuGroup operations (for local and remote)
- AddCpuGroup, ModifyCpuGroup
- Endpoint ManagementIPAddress
- CSTAActiveDevice operations
- GetCSTASMInfoRequest, GetCSTASMInfoResult
- JobAction GenericAuditHomeDN
- QueryPFRRequest LocalPortExpression
- CfCFSIE
  - DetermineCallBasedOnTransferringNum,
  - RingDurationIntCall,
  - RingDurationExtCall (renamed from RingDuration)
- MobileInfo
  - CellDeviceNumber,

  - RouteToCell
- TrunkingLicensingInfo RemainingDays
- DynamicLicenseInfo RemainingDays
- ClientAccessLicensingInfo RemainingDays
- OSMOLicensingInfo RemainingDays
- SLESLicensingInfo
- GenericJobRequest, GenericJobResult
- EndpointAttributes
  - AllowUnregisterStaleRegistrations (renamed from Reserved3),
  - CFLoopExtended,
  - VideoCallAllowed,
  - Reserved5,
  - Reserved6
- QueryAliasRequest QueryAliasRequest
- CACPolicyVideoCodecsAllowed CACPolicyCodecVP8
- CreateCACPolicyRequest MaxVideoCodecBandWidth
- CACPolicyInfo MaxVideoCodecBandWidth
- ModifyGToGCACPolicyRequest MaxVideoCodecBandWidth
- GToGCACPolicyInfo MaxVideoCodecBandWidth
- GenericJobRequest, GenericJobResult
- GetContTrcInfoRequest, GetContTrcInfoRes, ActivateContTrcRequest, ActivateContTrcRes, DeActivateContTrcRequest, DeActiveateContTrcRes,
- StartFileExportRequest, StartFileExportRes, StopFileExportRequest, StopFileExportRes
- Removed obsolete info for classic import/export format

3 Design Considerations

The following section lists and defines all of the assumptions that this solution is predicated on and limitations imposed by this design.

3.1 Provisioning Traffic Volumes

Once the system is populated with subscribers, provisioning traffic is relatively light (as compared to call traffic). Reports and high volume (bulk) activities can be scheduled using customer facilities and can be performed during off peak periods.

3.2 Sizing

The number of supported subscribers is a function of the hardware platform selected and the expected traffic volumes. The SOAP interface does not limit the capacity of entities.

3.3 System Requirements

The SOAP Server executes on both nodes of the OpenScape Voice system. Since provisioning traffic is expected to be low and bulk activities will be done during off peak periods, it is not expected to place a burden on call processing. This process will execute at a lower priority. No additional hardware is required.

3.4 Command Line Interface (CLI) Access

Subscriber profiles are NOT accessible via CLI.

3.5 Security

The SOAP Server is an open server behind a firewall. All requests will come from a Customer Relationship Management (CRM) system. There is no direct access to this provisioning interface (by subscribers or users) from outside of the system. The SOAP interface provides a security control to limit requests to all IP addresses or only those requests that come in on a specific IP Address.

If desired, IPSec will be used at the system level. The use of IPSec is transparent to the SOAP interface.

3.6 Requirements for Using this Interface

A web server or Customer Relationship Management (CRM) system is required to provision and maintain subscriber profiles. The CRM solution provides order entry, history, and other operator interface tasks. It will integrate with the Number Management System, Facilities Management System, as well as Billing System. It also will implement scheduling of subscriber provisioning actions.

The OpenScape Voice system is a live call processing platform; it does not provide any CRM features. Section 3.6.1, “CRM Required Features” lists features that are required to be provided by CRM systems.

3.6.1 CRM Required Features

- Order Entry Screens to operators to provide:
  - Operator authentication and controls (which operators can interact with which number ranges, which switches, or domains)
  - Auditing and history measurements of operator actions
  - Scheduling and resource reservations for scheduled activities (end of month and off-peak bulk provisioning)
- Interface to Number Management System to provide:
  - Valid number assignments (correct NPA, reserved numbers, premium numbers…)
  - Sterilization of old numbers (6 month reuse period)
- Interface to Billing Management System to provide:
  - Account number assignment and Directory Number correlation
  - Calling and rate plan assignment
- Interface to Facilities Management and Trouble Ticketing Systems to provide:
  - Tracking of resource allocation.
  - Subscriber problem (outage) handling and resolution

3.6.2 Required Interface Support

SOAP/XML is defined as a variable messaging interface. It is a tag/value pair interface much like TL/1. The main difference, however, is that in SOAP, the standard is that all tags are optional and new unknown tags can be ignored. The OpenScape Voice system uses this “non-strict compliance” convention to provide forward and backward compatibility.

Attention: There is no error condition returned to the client if tags are mistyped, misspelled or sent with incorrect case. These are treated as unknown tags and ignored.

3.7 Standards Support

OpenScape Voice supports WSDL 1.1, SOAP 1.1 and SOAP 1.2 over HTTP 1.0 and 1.1 supporting Doc/Literal and RPC encoding.

3.8 Interface Version Support

OpenScape Voice ignores unknown or mistyped tags and has defaults for new tags, and SOAP clients that use this interface are expected to do likewise. This will also facilitate communication with multiple OpenScape Voice systems that may have different software versions installed.

The OpenScape Voice system supports a GetVersion Interface that can also be used; the user of this interface may know what interface methods are available on the system. The OpenScape Voice system does ***not*** support a dynamic Application Programming Interface (API) discovery. See Section 5.1, “Interface (WSDL) Versions”, on page 61 for WSDL versions available on different OpenScape Voice system versions. See Section 7.1, “Get Version”, on page 94 for the method to retrieve the WSDL version from the OpenScape Voice system.

Some ***GET*** methods support a WSDL version, which limits the returned parameters to what was originally supported in the provided WSDL version. NOTE: This parameter will be obsolete in the future as SOAP clients are expected to ignore unknown tags.

Where interfaces have changed between versions, the OpenScape Voice system will mark the interface as deprecated (in this document), but continue to support that interface for one more release. The OpenScape Voice system supports the current release plus the two previous release’s interfaces to allow time for customers to migrate to the new interfaces.

4 Concepts

The OpenScape Voice system is versatile and has many elements that need to be provisioned. There are many different types of subscribers on the OpenScape Voice system supporting many different protocols and methods of connection. The following subsections describe some objects and concepts to be familiar with when using the provisioning interface.

4.1 Subscribers

Subscribers may be provisioned for any HomeDN previously defined on the system. Subscribers are all identified by their DNs. All subscribers are identified by a fully qualified DN.

The DN may be an internal DN only (fictitious). Fictitious DNs are real DNs on the platform. The user will need to set up the E.164 routing such that these DNs are NOT routed to.

Subscribers may be Residential or related to a business group (Business Group Lines). There is no real difference in the subscriber definition and only some of the services that can be assigned to that subscriber.

4.1.1 Profile Only Subscribers

In general, subscribers have a physical phone. When creating a subscriber, that physical phone information is provided and the endpoint/line is created and matched to the subscriber's DN. The OpenScape Voice system supports Profile Only subscribers. These subscribers are created without a physical line. Obviously, these subscribers cannot receive calls directly, but they can be used as the Pilot DN for an Hunt Group, for Remote Call Forwarding, as a Hot Desking subscriber, etc.

4.2 Feature Profiles

Feature Profiles are templates designed to simplify the management of subscribers. Subscribers may be managed independently with features applied directly to the individual or FPs may be used. Features may be managed at a FP level and then subscribers may be assigned to a FP. The features from the FP will be applied (cascaded or Inherited) to the subscriber. The cascading is performed as a background JOB within the OpenScape Voice system.

Features may still be applied directly to individual subscribers and those direct subscriptions override features from the Feature Profile. Overrides may be performed for an entire feature or for just a specific attribute of a feature. Subscribers who have Feature Profile features and attributes are subject to change when those values are changed in the feature profile.

For example, if the feature profile has CFDA and the subscriber is assigned to the Feature Profile, the subscriber will inherit the CFDA feature with its current FP settings. If the feature is active, the subscriber would have his/her phone forwarded to the CFDA destination defined in the feature profile. If the settings are changed in the Feature Profile, that subscriber will have his/her settings changed unless he/she (or the administrator) has changed the specific settings. A subscriber may change his/her specific settings for a feature either via the OSV Assistant or via the phone. A reset option is provided to allow the administrator to override local settings and reset those settings back to the feature profile. An example where this could be used is where an administrator wants to forcibly set all subscribers CFDA destinations to an attendant.

An administrator will have the ability to force change feature settings using the ResetLocPrecence option of the UpdateFeatureProfile interface. That option will overwrite specific local subscriber feature data and reset those values to be the same as the Feature Profile. Features that are provisioned directly to the subscriber are not affected by the reset option or by feature profiles.

An inherited feature *cannot* be deleted at the subscriber level. To disable a specific feature, it may be provisioned as **Denied** directly to the specific subscriber. The user can provision the UpdateSubscriberFeature and the PresenceFlags to reset the data values of a specific subscriber feature back to the definitions of the feature profile.

Services may not be provisioned against the Feature Profile as **Denied**. The service status of Inherit is a read-only value that indicates that the subscriber has the feature because it was provisioned to the subscriber's Feature Profile.

See Chapter 10, "Features" for a description of the service features that may be assigned to individual subscribers. Table 1 defines the features that can be assigned at the Feature Profile level. Since some features are dependent upon or conflict with other features, it is possible to assign features at the subscriber level whose prerequisite feature is assigned at the Feature Profile level. In this case, a problem could arise if the operator attempts to delete the prerequisite feature from the Feature Profile.

During Job processing, any dependent features found at the subscriber level will be removed if the pre-requisite feature was removed from the Feature Profile and any conflicting feature assigned to the FP will not be inherited by a subscriber who has a conflicting feature assigned directly.

A warning will be included in the Job Result data to indicate this action. Feature interactions are fully listed in Section 1.2, "Subscriber Interactions", on page 23. Table 1 is a summary of feature dependencies for which the scenario described

here may occur. If a feature in the Feature B column is removed from the Feature Profile, the corresponding Feature A (if provisioned directly to the subscriber) will also be removed.

Feature A	Feature B
CIDCW	CND or CNAM
CCW	CW or CIDCW
RACF	CFV
RFA	SRS

Table 1 BG Feature Dependencies

If an update to the BG would cause conflicts to occur to the BGL (for example CT), the entire update to the BGL will fail. For example, multiple features can be applied in one request. That request may contain CFV and CT. This may be accepted by the Feature Profile, but the propagation to specific subscribers may fail.

FPs may be Residential or Business Group. They are treated identically and separated only for better management because of the various types of services between Business Group and Residential subscribers.

Refer to Chapter 14, “Feature Profiles” for more information regarding parameters and management.

4.3 Branch Offices

A branch office is a group of related subscribers or/and endpoints, and it is a subentity of a business group. All subscribers and endpoints of a branch office are linked to the branch office via their associated endpoint attribute, which matches the representative endpoint recorded for a branch office. The representative endpoint is the endpoint of a proxy with the 'Survivable' endpoint attribute set to true.

Subscribers and endpoints in a BG that were administered under the numbering plans are now part of a hierarchical concept, where they belong to one of the branch offices (if associated with a representative endpoint) or to the “Main Office” (either not associated to an endpoint or associated to an endpoint that is not a representative endpoint of a branch office).

The following data are administrable per branch office:

- Name: Each branch office shall receive a unique (per BG) name that can be used.
- Representative endpoint: The representative endpoint is the endpoint that is configured as the associated endpoint for all subscribers and endpoints of the branch office. With this association, the subscribers and endpoints of the branch office are logically tied together. For branch offices with a survivable

proxy, following the proxy registration model, the representative endpoint is the configured survivable proxy endpoint. For branch offices with a survivable gateway, following the dual registration model, the representative is the configured survivable gateway endpoint.

- Default office code: Each branch office receives a default office code upon creation. It is however possible to assign subscribers from other office codes for the same branch office.
- Default PNP: Each branch office receives a default private numbering plan upon creation. It is however possible to use multiple private numbering plans for the same branch office.
- Default Routing Area: Optional field that indicates the default routing area of a branch office. Again, multiple routing areas can be used per branch office.
- DID pool for correlation service: This can be created only if the branch office is a survivable branch office (indicated via the 'Survivable' attribute on the branch office's representative endpoint).
- OpenBranch indicator: Optional attribute that indicates that the representative of this branch office is a HiPath OpenBranch device for which administration will be supported through the Common Management Portal.
- Username: The username that will be used by the OpenBranch Assistant to administratively log on to the OpenBranch device. This username must be supplied when the OpenBranch attribute is provided.
- Password: The password that will be used by the OpenBranch Assistant to administratively log on to the OpenBranch device. This password must be supplied when the OpenBranch attribute is provided.

All numbers of the DID are configured as a vertical service in the PNP used by the SIP gateway of the survivable branch office. The vertical service is linked to the correlation (TIE) service.

4.4 Recent Change Logging

This capability provides security audit trails and recent change logs on the OpenScape Voice system via the OpenScape Voice Assistant. Also known as Security Event Logging, this functionality allows the recording (in a log file) of all provisioning activities that are performed on the OpenScape Voice system.

4.4.1 Recent Change Logging in the SOAP Server

Whenever a SOAP request is received that involves the creation, deletion, modification, or retrieval (e.g., for display/view) of data on the OpenScape Voice system, the Event Logging API function RtpSecEvtSendChangeLogEvent provided by RTP is called to log the event. The data passed to the API function consists of:

- The operator ID value passed in the SOAP request.

  Note: In the case of certain retrieval requests that do not include an operator ID in the interface, a string with the IP address identifying the system that sent the request will be used here.

- A block of data encoding the content of the request:
- The name of the SOAP request (for example: CreateSubscriber, UpdateSubscriberFeatures, and so on).
- The identity of the object being created, deleted, or modified (eg: for a Subscriber, the DN; for a Business Group, the BG Name; for a Feature Profile the pair FP[:BG] name; etc.).
- A detailed description of all parameters included in the request, identified explicitly with the names of the parameters and their values. Parameters not included in the request or sent with no value may be omitted.
- An indication of success or failure of the SOAP request.

The detailed description of the request will be the incoming XML message after stripping off everything that is not needed (end tags, empty tags, header, and so on. All passwords and PINs in the incoming SOAP message will be included.

All functions that handle SOAP requests collect the data necessary to log the operation (operatorId, operation, resource) in the beginning of the function and call the function setChangeLogParameters. The log function is called after the arguments that will be passed to the function are validated and before the call is made to any other function. This function writes the data received into a structure. The call to the API is only done when the SOAP response corresponding to the logged request is sent. This approach ensures that only one log entry is generated for each SOAP request.

```
int setChangeLogParameters ( const char* operatorId,
                             const char* operation,
                             const char* resource)
```

The following text describes an example scenario that demonstrates how a SOAP request is logged. In the example, the operator “admin” creates a Business Group called “bg1”.

The function soap_recv_callback, that currently traces the incoming SOAP request, calls the function processIncomingRequest. This function determines whether the incoming message is a new message or a continuation of the previous one, by verifying the beginning of the string. If it is a new message, clears the incomingSoapRequest and writes the new incoming message. Subsequent messages are appended to incomingSoapRequest. The function processIncomingRequest copies the incoming XML string into the incomingSoapRequest, stripping off everything that is not needed (end tags, empty tags, header, and so on).

The function unsp__CreateBG2 corresponding to the SOAP request, calls the function setChangeLogParameters and provides the parameters required to be logged:

```
setChangeLogParameters (OperatorId, (char*)
"CreateBGRequest", BGName)
```

The function setChangeLogParameters populates the structure with all parameters except the result field which is provided by setErrInfoNExit. The first function called (unsp__CreateBG2) is the one corresponding to the operator action. This function sets the flag isFirstFunctionCalled. Other functions unsp__CreateBG2 may call aren't logged, because the flag is set. In this example, when the request processing finishes, the structure logDataCollection will have the following values:

- logDataCollection.operator = "admin"
- logDataCollection.operation = "CreateBGRequest"
- logDataCollection.resource = "bg1"
- logDataCollection.result = RtpSuccess
- logDataCollection.isFirstFunctionCalled = TRUE
- incomingSoapRequest [] = "<BGName>bg1<OperatorId>admin <CreateBGRequest><DisplayNumber>3059234090"

When the SOAP response is sent, the function soap_sent_callback, used to trace the outgoing message, also resets the flag isFirstFunctionCalled, calls the API and provides the log values. The API is defined as follows:

```
RtpSecEvtSendChangeLogEv  &logDataCollection.operator,
ent(                      &logDataCollection.resource,
                          &logDataCollection.operation,
                          logDataCollection.result,
                          incomingSoapRequest)
```

An event will be logged using the OpLogReport API if the return code from the Security Event Logging API is not successful, indicating failure to log the entry.

4.4.2 Recent Change Logging in SOAP Mass Provisioning and SOAP Export

Mass Provisioning and Export operations are also logged, in the same general manner as described above for the SOAP server, but the format of the logged data is different.

For SOAP Mass Provisioning, the comma separated string representing the input mass provisioning command from the input file is logged.

For SOAP Export, the operation performed is always the retrieving of data and thus the name of the operation is logged with the generic name: soapExport

The identity of the objects being exported are as specified in the input command, defined by the operator (eg: the StartDN; the BGName, etc.). In the case of exporting a list of DNs, the resource logged will be the one given by the –startDn argument.

The entire string entered by the operator to execute the command is logged, including all arguments passed to the application, identified explicitly with the name and the value, to describe with detail the operation being performed.

Note: As a result of Event Logging requirements, both SOAP Export and SOAP Mass Provisioning are invoked from within the RTP CLI Expert Mode. This becomes the normal method of performing mass provisioning, since this enforces the requirement that the user be logged in when performing provisioning operations. The RTP CLI Expert Mode interface to soapMassProv and soapExport includes a password parameter, as well as the user ID of the user logged into the RTP CLI.

It is still possible to invoke both executables from the command line for testing purposes. This is expected to be used only internally and requires knowledge of a password defined in an RTP parameter that is not customer-accessible.

4.5 Hunt Groups

Hunt Groups allow calls to be routed to an idle line within a group of specified lines. Each individual line must have its own private DN. A call is placed to a Hunt Group by dialing the Pilot Number. Hunting does not occur when a non-pilot DN (i.e., a line's private DN) is dialed. When a call is placed to a Hunt Group, the attempt to terminate the call begins with the line (i.e., member) designated by the Pilot Number based on the group's hunt method (see Hunt Algorithms below). When all lines are busy, overflow treatment is provided to a queue, overflow DN, or to an intercept treatment. The lines in a single Hunt Group cannot be a mixture of different Business Groups nor can they be a mix of Business Group and non-Business Group lines. Refer to Chapter 15, “Hunt Groups” for more information.

4.5.1 Types of Hunt Groups

Two types of Hunt Groups are supported. Both are routed to by dialing a Pilot DN.

- Pilot Hunt Group (AKA, Profile-Only Hunt Group): In this type, the Pilot DN points to the Hunt Group, and line 1 of the Hunt Group has a DN that is different from the Pilot DN.
- Master Hunt Group: In this type, the Pilot DN of the Hunt Group is also the DN of the first line in the group. Line 1 of the group is also the “master” of the group, meaning that it can administer the various features assigned to the group. From V7 and upwards system restricts pilot allowed position to 1 only, and members are restricted to the remaining position numbers.

4.5.2 Hunt Algorithms

The following hunting algorithms are supported:

- Sequential: An available line is searched for starting with line 1 of the group sequentially through the group until the last member is reached. The first available line is chosen to present the call to.
- Circular: An available line is searched for beginning with the line position stored when the previous call to the Hunt Group was made. When a line is selected to complete a call to the group, the line that is one past it in the group is marked to become the starting point for the hunt on the next call to the Hunt Group. For example, if the last line in the group was chosen for the previous call, then this is the first line in the group for the next call.
- Uniform Call Distribution (UCD): The line selected is the line which has been idle for the longest time.
- Manual: When the Hunt Group hunt selection method is Manual, the distribution of calls to agents will not be done by the OpenScape Voice system. All incoming calls will be queued. For the distribution to work, the Hunt Group should also be marked for CSTA, allowing an external application to be notified of calls going into the queue, and to subsequently retrieve (reroute) those calls.
- LinearUnconditional: Obsolete / No longer supported
- Parallel: All available members of the Hunt Group are alerted simultaneously whenever a new call arrives at the Hunt Group.

4.5.3 Blocking of DNs

When the Pilot DN of the Hunt Group is dialed, it is the Blocked status of the Pilot DN that is checked when determining blockage, not those of individual members. If the Pilot DN is blocked, hunting will not occur. If it is not blocked, the normal hunting will occur, and will not consider the blocked status of an individual member.

When a member's DN is dialed directly, it is the Blocked status of that DN that is checked when determining blockage.

4.5.4 Traffic Measurements

When the Hunt Group is classmarked to maintain traffic measurements, the following measurements are maintained:

- Number of Incoming Calls to the Hunt Group - Count of the number of incoming calls which are attempting to terminate to the Hunt Group (i.e., which attempt a hunt).
- Number of Outgoing Calls from the Hunt Group - Count of the number of outgoing calls originated from the members of the Hunt Group.
- Number of Overflow Calls - Count of the number of terminating call attempts to the Hunt Group that are unable to initially complete to a line in the Hunt Group because an available line is not found.
- Total Usage of the Hunt Group - Measurement of the total Hunt Group usage for all calls (incoming and outgoing) to the Hunt Group. For incoming calls, the usage measurement begins when the call is answered by a line in the Hunt Group. For outgoing calls, the usage measurement begins when the call is answered by the party called by the line in the Hunt Group. In both cases, it is stopped when the call is released. It is recorded in seconds.
- Queue Attempts - Count of the number of attempts to place a call in queue. It records both successful and unsuccessful attempts.
- Queue Usage - Measurement representing the total usage for all calls while in queue. It is recorded in seconds.
- Queue Overflow - Count of the number of attempts to place a call in queue that failed because the queue was full.
- Queue Abandons - Count of the number of calls that were abandoned by the originator of the call while in queue before being connected to a member of the hunt group.

A subscriber or BGL can be a member of multiple Hunt Groups. In this case, one Hunt Group is designated as the primary Hunt Group. If subscriber S is a member of Hunt Group A and is also the Pilot DN of Hunt Group B, hunting through Hunt

Group B takes place only if subscriber S is called directly, not if the call is presented to subscriber S as a result of hunting after an incoming call to Hunt Group A's Pilot DN. When a subscriber or BGL is a member of multiple Hunt Groups and its primary Hunt Group is removed, the member's primary Hunt Group designation is updated to refer to one of the still existing Hunt Groups (the assignment will vary).

The Hunt Group PilotDn and all of the terminals must be created as subscribers before being turned into an Hunt Group Pilot or Hunt Group terminal. The subscriber can have a connection (phone). If the Hunt Group Pilot subscriber has a phone, then that phone is the master phone for the Hunt Group and controls the Hunt Group services. If the subscriber does not have any connection info, then the Hunt Group is just a container with its service features.

Service features can be applied to the Hunt Group PilotDn or an Hunt Group Member (Dn). The assignment of service features to pilot or terminals is performed through the modification of file featurelist.dat. Each line of this file provides information for each feature. By setting "mlhg_pilot" column 'true', the feature is applicable to a pilot DN. A new column named "mlhg_member" is inserted for allowing or denying a feature to a terminal.

When a caller calls a subscriber that is an Hunt Group PilotDn, the services on the PilotDn — not the member or terminal to which the call actually hunts and terminates to — are executed. When a caller calls a subscriber that is a member of one or more Hunt Groups, the subscriber's services are executed and not those of any Hunt Group that the subscriber may be a member of. An exception is Keyset Operation. A subscriber can be a keyset phone and also be a member of a Hunt Group. In this case, it is desirable that when an incoming call reaches a keyset Hunt Group member via hunting, the call should be presented to all devices where the subscriber has a line appearance. To accomplish this, it is necessary for the Hunt Group service to start the keyset service on each individual Hunt Group member, if it is a keyset subscriber, and also to stop the service if the No Advance timer expires and hunting must continue.

Outgoing calls always invoke the subscriber's (Hunt Group member) features.

Currently, an Hunt Group member must have a DN. It can be marked non-external if it is a non-public number. When an Hunt Group line originates a call, if the phone has an E.164 DN, that DN will be sent as the calling party number, along with that DN's privacy indication. Otherwise, the Pilot DN is used, along with the DN's privacy indication.

The following limits apply to Hunt Groups in the OpenScape Voice system:

- Maximum of 25,000 Hunt Groups per OpenScape Voice system
- Maximum of 2,048 members per Hunt Group
- A member can be in up to 32 different Hunt Groups

4.5.5 Hunt Group Restrictions

Some parameters of the Hunt Group are accessed and controlled using the new Hunt Group methods. Others may be referenced using the existing subscriber methods. The following restrictions apply to subscribers that own a Hunt Group.

- An Hunt Group owner (subscriber) cannot be removed or disconnected until the Hunt Group is removed.
- The RCF service feature cannot be applied to the Hunt Group owner until the Hunt Group is removed.
- The SRS service feature cannot be applied to the Hunt Group owner (Pilot DN) until the Hunt Group is removed.
- An Hunt Group owner (subscriber) may not be exported or have the DN changed.

Hunt Group phones can be lines with E164 numbers, BG private numbers, or non-numbered terminals. The phones are identified by their service_id. This is the E.164 DN (if it's defined), the BG private DN (if there is one), and, lastly, by PilotDn-TermNumber.

In these releases, all subscribers are identified by a DN. NonDNHunt Group terminals are not yet supported. The External DN flag originally introduced to indicate whether BGLs are reachable only from within the BG, however, is used by the Hunt Group service for this purpose. If a non-DN terminal is needed, a fictitious DN can be used at subscriber creation and the External DN flag set to false.

4.5.6 Hunt Group Traffic Measurements

The Hunt Group provisioning commands (See Section 15.2, "Hunt Group Phone Services", on page 621) include a parameter called TMStatus, which allows the operator to enable or disable traffic measurements for a particular Hunt Group. When enabled, traffic measurement data is generated by the OMM process for that Hunt Group.

In general, when traffic measurement data is generated, a subdirectory with the format *SRX3000-YYYYMMDDTHHMMSS* is created in the root traffic measurement directory (`/global/user/oms`). These same directories are used for Hunt Group traffic measurement data.

The names of these directories can be retrieved using the *GetMlhgTMData* SOAP request (See Section 15.2.9, "Get Mlhg TM Data", on page 643). This request takes as input a single subdirectory name in the format mentioned above. When provided, this indicates that only subdirectories that were created after the timestamp of that subdirectory are returned. Also, since these same

subdirectories also contain other non-Hunt Group traffic measurement data, the files are checked and the subdirectories are returned only if at least one Hunt Group traffic measurement file (any file beginning with the characters “mlhg”) is present.

The returned directories consist of the full path to the subdirectory that contains Hunt Group traffic measurement data, in the form: */global/user/oms/SRX3000-YYYYMMDDTHHMMSS*.

The actual generation of the Hunt Group statistics files on the OpenScape Voice system must be enabled via the OpenScape Voice Assistant or CLI using the Operation Measurement Management (OMM) settings.

4.6 Jobs

The OpenScape Voice system processes some provisioning actions as background jobs. In future releases, it may be possible to schedule actions, such as scheduling the disconnection of a subscriber (likely to occur at the end of the month) to occur later on the OpenScape Voice system. At this time, however, jobs are used for SOAP requests that require more processing time than a client typically will wait for.

The most common type of request that requires more time than available is an Update to a Feature Profile (FP), which may require the updating of several thousand subscribers. Scheduled job processing is also used for this operation.

Upon receiving this type of request, the OpenScape Voice system returns the JobId tag in the ResultCodeStruct. When this field is returned, it means that the request has been successfully received, validated, applied to the FP, and updates have been scheduled for the affected subscribers/BGLs.

The status of the job may be obtained using the GetJobInfo method and a list of existing jobs may be obtained using the GetJobList method. The CancelJob method may be used to terminate a running or waiting job.

Refer to Chapter 17, “Background Job Processing Interfaces” for more information.

4.7 Media Gateways and Servers

Attention: The OpenScape Voice system supports the RG 8700, HiPath 4000, and the Mediatrix media gateway’s. These are all SIP endpoints and are therefore not configured using the “Create Media Gateway” SOAP commands. For the

OpenScape Voice system, the only Media Gateway that can be created using this command is a Media Server (supported: OpenScape Media Server, Convedia Media Server).

4.8 Calling Number Display Modification

From the end user perspective, it is useful to see the number being displayed that can be used exactly to call back the calling party (instead of prefixing the public access code, country code, etc. to the display number before calling back). In order to support the display of a dial-able number, there are administrative tables for number modification.

The dial-able display number feature is per numbering plan based. It is applicable for all markets, all solutions. All of these display number related tables are optional.

Any existing features or future features which have the number display functionality shall interact with the dialable display number feature. The dialable display number feature applies only if the display number supporting tables are provisioned. If they are not provisioned, then the current display number function will be used with no modification applied. To disable the dialable display number feature, the entries in these tables need to be removed.

In V3, the Numbering Plan Identifier (NPID) and Type Of Network (TON) parameters are divided into 'input' and 'output' values. The {input, output} values for NPID are known as 'fromNPID' and 'toNPID', respectively. The {input, output} values for TON are 'TONin' and 'TONout' respectively.

When the TONin value is unknown, the ten-digit calling party number is compared to the Number Definition Table and the NOA value is used as the TONin.

After the Modification Table is searched, the 'toNPID' and 'TONout' values are used in the search of the Prefix Table. The result is to find the PNAC & Prefix for the displayed number.

The Modification Table is now indexed by the three values: fromNPID, toNPID, TONin. In addition, the OVERRIDE flag has been removed from the Modification Table in V3. The output from the Modification Table is the TONout and the OPTIMIZE flag.

The following subsections describe the tables and their contents. For information on the SOAP/XML provisioning capabilities to support Number Modification functionality, please refer to Chapter 13, “Number Modification”

4.8.1 Display Number Definition Table

The Display Number Definition table shows the breakdown of the Number Plan. It defines the display number table by providing the administrator the overview of the Numbering Plan attributes across the system. The table contains the Numbering Plan data for Country Code, Area Code, Exchange Code, Skip Digits, Unique Level for E164, and Level 2, Level 1, Level 0, Unique Level for PNP.

There can be multiple entries for the same Numbering Plan because a numbering plan can service one or more area codes and office codes at the same time. There are no unique fields to the table. Only creation and deletion of entries are supported for this table; no modification is supported.

An entry in the display number modification table that has a -PRE tonOUT defined which is of the same level as the tonIN, i.e. NATIONAL to NATIONAL-PRE that also has optimization turned on, should return a prefixed number even if it cannot optimize the number, if there are no matching entries in the display number definition table.

The format and valid range of the **Country Code (CC), Area Code (AC), Exchange Code (EC/LC),** and **Location Code (L2, L1, L0)** are controlled by RTP parameters.

Location Code – A digit string of {0-9} of max 14 digits represents the location code.

The locCode consists of L2, L1 and L0 codes. The SOAP/XML interface requires the entry of a single string and the lengths of the L2, L1 and L0 portions.

L1 and L2 are optional.

L1 cannot be administered without L0.

L2 cannot be administered without L1.

locL0Len - L0 code length, an integer {1-4}.

locL1Len – L1 code length, an integer {0-6}.

locL2Len – L2 code length, an integer {0-4}.

Skip Digit (SK): This parameter specifies the number of digits which are to be skipped to extract the extension number when extension number and node access code overlap (i.e. 923-1790 with skip digits = 2 leads to extension number 31790). It is optional. The default value is 0.

Unique Level (UL): This parameter specifies the lowest unique level. Lowest level, which makes a called party unique. This is the extension portion of the number (when using PNP) or the subscriber number (when using E.164). This is a read-only field.

It is not be possible to add PRIVATE entries to this table if no PUBLIC entries exist. This is to ensure that conversion between PRIVATE and PUBLIC will work. Also you cannot have a lower level of number definition on the PNP side of the display number definition without the same level of definition on the E.164 side. Otherwise, it is not possible to determine the extension component of the number before converting it to PNP

Table 2 shows an example of the data contained in the Display Number Definition Table.

Numbering Plan	**ISDN (E.164)**		**Private (PNP)**	
	CC **AC** **LC** **UL**	**SK**	**L2** **L1** **L0** **UL**	**SK**
Munich NPID=1	49 89 722 SUB	3	 722 EXT	3
Boca NPID=2	1 561 923 SUB	2	 96 EXT	2
Global NPID=9999	1 408 492 SUB			
SK = Skip digits, i.e. Number of digits which are to skipped to extract the extension number when extension number and node access code overlap (i.e. 923-1790 with skip digits = 2 leads to extension number 31790). Default value: length of the LC or L0 (i.e. no overlapping)				
UL = lowest unique level. Lowest level, which makes a called party unique. This is the extension portion of the number (when using PNP) or the subscriber number (when using ISDN).				

Table 2 Display Number Definition Table

The provisioning of the Display Number Definition table is optional. If there are no entries to the table, number modification will not take place. The validity of the entry is based on the administrator's knowledge of the numbering plan; no cross reference checking is done during the provisioning.

4.8.2 Display Number Prefix Table

The Display Number Prefix Table constitutes the default network access code and number prefix code per Numbering Plan for displaying the dialable numbers. It is provisioned statically by the administrator with the knowledge of the effective numbering plan information.

There will be only one set of default access code per numbering plan. The network access code and the number prefix code are provisioned based on the Numbering Plan and the Type Of Number (TON). They are used for prefixing the number to be displayed.

The key to the table is the combination of originating Numbering Plan Name (FromNpId) and the Type Of Number (TON).

Numbering Plan Name (FromNpId): This is the numbering plan ID of the subscriber who originated the call. It can also contain the value NIL, meaning ANY numbering plan.

Type Of Number (TON): The TON parameter defines the conversion of an explicit “calling number” to a desired format before the number is transmitted to the destination. It is mandatory.

The possible values:

INTERNAT – International number;

NATIONAL – National number;

SUBSCR – Subscriber,

LEVEL2 – Location Level 2,

LEVEL1 – Location Level 1,

LOCAL – Local subscriber Number

Public/Private Network Access Code (PNAC): The PNAC parameter describes the public or private network access digit. For example, the 9 for the public off-net access, or 8 for on-net access code. The format and valid range is controlled by RTP parameters. It is optional. The default is an empty string.

Prefix: The Number Prefix parameter is the prefixed digits for the particular call type. For example, 00 for international call; 0 for national call. The maximum length is 15, the format is controlled by the Srx/Xdm/DnCode/digits/charSet in the SrxDigitLen.parm file. It is optional. The default is an empty string.

Table 3 shows an example of the data contained in the Display Number Prefix Table.

From NPID	TON	PNAC	PREFIX
1	LEVEL2	8	1
8	LEVEL1	8	
3	LEVEL2	8	1
ANY	LEVEL1	8	

Table 3 Display Number Prefix Table

The provisioning of the Display Number Prefix table is optional. If there are no qualified entries for the Numbering Plan, then no digits will be prefixed to the display number. The validity of the entry is based on the administrator's knowledge of the numbering plan; no cross reference checking is done during the provisioning.

4.8.3 Display Number Modification Table

The Display Number Modification Table provides the default number display format between originating NP and terminating NP.

For the case that the TON is unknown during a call setup, this table shall provide the desired display number format for the calling party based on the originating NP and termination NP. For the case in which the number format is known by call processing software, it is not necessary to query this table.

The combination of the Originating Numbering Plan Name (fromNPID) and the terminating Numbering Plan (toNPID) is the key to the table. The pair must be unique, i.e. it cannot have two entries with the same directional relationship between two numbering plans.

Originating Numbering Plan Name (fromNPID): This is the numbering plan ID of the subscriber who originated the call. It can also contain the value NIL, meaning ANY numbering plan.

Terminating Numbering Plan Name (toNPID): This is the numbering plan ID of the subscriber terminating to. It can also contain the value NIL, meaning ANY numbering plan.

Type Of Number (TON): The TON parameter defines the conversion by call processing of an explicit "calling number" to a desired format before the number is transmitted to the destination. It is mandatory.

The possible values:

- INTERNAT – International number;
- INTERPRE – International number with prefix;
- NATIONAL – National number;
- NATIOPRE – National number with prefix;
- SUBSCR – Subscriber;
- SUBSCRPRE - Subscriber number with prefix;
- LEVEL2 – Location Level 2;
- LEVEL2PRE - Location Level 2 with prefix;
- LEVEL1 – Location Level 1;

- LEVEL1PRE - Location Level 1 with prefix;
- LOCAL – Local subscriber Number
- LOCALPRE - Local subscriber Number with prefix
- ALL - all numbers; applicable to TONin
- ANY - any number; applicable to TONin and TONout
- ANYPRE - any number with prefix; applicable to TONout

The parameter values for TON depend on the particular tonIN field and have an effect on the format of the “calling number” display.

Privacy Type: This parameter indicates the privacy setting for specific call scenario using the numbering plan. However, a user can have his own privacy setting based on some feature assignment. The effective privacy setting to be applied is the more restrictive rule in combination with NP privacy setting and the subscriber’s privacy setting. The possible values: Transparency, Blocked, BG Number. Default: Transparency.

Table 4 describes each of the possible Privacy Types.

Values of Privacy Type parameter	
TRANSPARENT	The privacy settings remain unchanged (transparent).
BGDISPLAYNUM	The display number for the Business Group is used as the calling party number.
BLOCKED	The station number is blocked and will not be transmitted

Table 4 Privacy Type Parameter Values

Optimized: This parameter indicates that the display number is to be further modified to an optimized format. The possible values: True (Yes), False (No). The default value: True. When it is set to True (Yes), then the numbers of the A and B parties will be compared after modification and optimized to remove any similar component parts.

Table 5 shows an example of the data contained in the Display Number Modification Table.

From NPID	To NPID	TON	Privacy Type	Optimized (Y / N)
1	2	LEVEL2	BLOCKED	N
8	2	LEVEL1	TRANSP	N
3	2	LEVEL2	TRANSP	Y
ANY	3	LEVEL1	TRANSP	Y

Table 5 Display Number Modification Table

4.9 Business Groups

A Business Group (BG) is a group of related subscribers. Subscribers may optionally belong to a Business Group Line (BGL). BGs present new service features to BGLs as well as an easier method for managing BGLs.

BGs have a status, main numbers, attendant numbers, message detail recording (MDR) information, a private numbering plan, and service access codes. Some features can be assigned to BGs.

If no Numbering Plan Name is provided when a BG is created, the BG will use the system default (E164) Numbering Plan. Business Groups *cannot* share Private Numbering Plans.

Once a BG is created with a numbering plan, it can be changed to another numbering plan if there are no subscribers existing in the BG.

When creating a BG, the system generated Customer Id is used as the Business Group Index. The user-specified Business Group Name is used as the Customer Name.

4.10 Business Group Department and Services

Business Groups can be divided further into departments to which Business Group Lines can then be assigned. Departments can be used to control the display information for the various callerId service features.

A Business Group may have services. The following sections define the services that can be assigned at the Business Group level.

4.10.1 Main Numbers

A business group can have a maximum of 100 main numbers. The main numbers can be normal lines or virtual DNs that are not associated with physical lines. If the main number is virtual, it is related to an attendant number. On an incoming call to one of the virtual main numbers, the services will process this number and redirect it to an attendant. The call will ring with a distinctive ringing tone to inform the attendant that the call is from one of the main numbers.

A subscriber profile will be created for the virtual main numbers, but the subscriber will not be allowed services or any physical connections. The main and attendant numbers must fall within the allowable number ranges of the BG.

Internally, the HomeDn destination of the Main (Virtual) Number is set to MainNumberService. If the main number is a real DN, then you do not need to specify an attendant number.

The attendant numbers must be provisioned before you provision the main numbers (virtualDN). An auto-attendant number is the number for redirecting calls that terminate at the main number if the auto-attendant service is available. Auto-attendant availability is indicated by the auto attendant status flag. If the auto-attendant flag is set to true, then the auto-attendant number must be specified.

The attendant number, auto-attendant number, and auto attendant status flags can be updated for a particular main number entry.

When a subscriber who has only a private number makes a external BG call, the BG's DisplayNumber is presented as the calling number. The DisplayNumber is required when the BG is created.

4.10.2 Attendant Numbers

You can assign one or more Attendant Numbers to a Business Group. For example, a multi-location Business Group can have different Attendant Numbers that correspond to different locations.

To create an Attendant Number for a BG, update the BGL's BGAttendant Number Flag to **True** in the subscriber's Account management Info.

4.10.3 Message Detail Recording (MDR) Information

MDR information is provisioned on the business group to provide enough information in the Call Context call Detail Recording (CDR) to support the generation of Billing AMA Format (BAF) MDR modules that are appended to the BAF AMA base structure records, upon mediation of these CDRs for calls originating/terminating from/to Business Group facilities.

The MDR Customer Identification must be unique for each BG. The MDR information is applicable to the BG only and cannot be provisioned at the BGL level.

4.10.4 Business Group Subnet E911 LIN Administration

SIP phones connect to the OpenScape Voice system via TCP/IP. With TCP/IP, there is no location information available for the phone. E911 LIN Administration is provided to give the Business Group phone location capability.

Subnet IP address ranges can be defined to identify location information. When SIP Phones register, their locations can be determined by the router or hub address through which they connect to the network.

A phone connecting from an area not covered within the subnet addresses defined can enter a temporary LIN address during the initial phone registration.

4.11 Business Group Traffic Measurements

The BG Traffic Measurements feature is provisioned on a per BG basis. The feature is enabled by BgTM service. The measurements data will be collected and recorded as CSV files by the Operational Measurements Manager (OMM). The files will then be available to be pulled via FTP to OpenScape Voice Assistant. When a request from the Assistant for a BG measurements data, the BgTM service will return a list of measurement file names with the date information associated with each file and the directory name to the Assistant.

Administrative controls over data file retention periods and multiple collection intervals are provided by CLI/OpenScape Voice Assistant. Three different collection intervals are supported for BG measurements group. The Administrable Interval can be configured to one of the following durations: 5 minutes, 15 minutes or 30 minutes, Hourly Interval is the sum of all of the administrable intervals that occurred during the hour. Daily Interval contains the sum of all the hourly intervals that took place during the day.

The per BG data files will always be produced as daily files that can be accessible at any time throughout the day. Each BG's OMM data file shall contain all the interval measurements for that BG. For example, there will be 24 hourly entries, plus a 25th daily entry in each BG file.

The BgTM service is assigned to the BG only and cannot be provisioned at the BGL level.

Note: The actual generation of the BG statistics files on the OpenScape Voice system must be enabled via the OpenScape Voice Assistant/CLI using the Operation Measurement Management (OMM) settings.

4.12 Numbering Plan

Numbering Plans (PNPs) have replaced Dialing Plans. The Numbering Plan is provisioned as part of the Business Group data. If no Numbering Plan Name is specified, the default is provisioned as the system default Private Numbering Plan.

The system default Numbering Plan can only be provisioned and maintained via the OpenScape Voice Assistant or CLI. Only the system default Numbering Plan can be shared among different Business Groups. A Private Numbering Plan

associated with a BG cannot be shared by another BG. The PNPs override the E.164 Numbering Plan, but may also make use of the E.164 Numbering Plan to avoid duplicating that data.

In order to configure multiple private numbering plans within a single BG, it is possible to assign a Number Plan Name to a Business Group Line (BGL - Subscriber) or to an Endpoint Profile. If a BGL or EP is not assigned a Number Plan Name then the Numbering Plan associated with the BG is used.

By assigning the same Number Plan Name to group of BGLs, that group can make calls and access features using their common numbering plan.

4.12.1 Service Activation Codes

The service activation code table for Business Groups can be provisioned through the OpenScape Voice Assistant. The table contains the service feature activation codes that subscribers will enter to initiate service features, such as *69 for Recall. There are no default settings when the system is delivered and no service features can be operated until this table is configured.

The service feature activation code settings are configured on a per Business Group basis and are defined in the Numbering Plan section (Section 12.1, “Numbering Plan”, on page 516) through the SOAP interface.

4.13 Keyset Information

Subscribers and BGLs can be created with keyset information, if desired. This section discusses the keyset-related attributes that may be associated with a subscriber or BGL. There is also a keyset service that is applicable at the Business Group level, but this is assigned independently of the provisioning of keyset information on a subscriber/BGL basis and is only provided as a way to store certain data items at the BG level.

All of the following parameters apply to the *UpdateKeysetInfo* SOAP request (Section 9.12, “Update Keyset Info”, on page 356) and the KeysetUse parameter also applies to the *CreateSubscriber* SOAP request (Section 9.1, “Create Subscriber”, on page 303).

4.13.1 Keyset Use (enum)

This attribute describes whether a DN is to be used for Keyset Operation, and if so, in what way. The possible values are:

- ***KeysetPrimaryLine***: Indicates that the subscriber DN is to be used as the Primary Line for a keyset, where primary line is defined as the first line to register from a given SIP endpoint and the line used by default when no secondary line key is pressed.
- ***KeysetPhantomLine***: Indicates that the subscriber DN is to be used as a Phantom Line for one or more keyset where Phantom Line is defined as a line that does not appear as a Primary Line on any keyset, but may have an appearance as a Secondary Line on one or more keysets.
- ***KeysetNone***: Indicates that the subscriber DN is not to be used as for Keyset Operation and no additional keyset-related attributes will apply. This is the default value for KeysetUse if the parameter is not supplied on the *CreateSubscriber* SOAP request.

4.13.2 Manual Hold Timer (range 5..3600)

This timer is activated when the user places the line on manual hold. If this line is picked up on any device where this DN has an appearance, the timer is stopped. If the timer expires, the device where the call was placed on hold is recalled. The valid range for this timer is from 5 to 3600 seconds, and the default value is 40 seconds. This default value is always applied on the *CreateSubscriber* SOAP request, since this parameter may not be input at the time of creation. This parameter is only expected if the Keyset Use field is set to Primary or Phantom.

4.13.3 Line Appearances

This parameter is only valid for Keyset Primary Lines as it represents a list of DNs that can be associated with this DN. In other words, a list of DNs that is allowed to register at the SIP endpoint where this DN is the Primary Line. Up to 29 BGLs may be included as Line Appearances and each Line Appearance must be a valid DN (no more than 20 digits in length), representing an existing BGL that is a member of the same BG as the Primary Line. This means that it is not possible to configure a DN as a Line Appearance before the subscriber associated with the DN has been created. A line may appear as a Line Appearance on up to 40 devices. For more information, refer to Section 9.5, “Get Keyset Primary List”, on page 344.

Note: It is possible to retrieve a list of devices identified by the Keyset Primary Line on the device on which a particular keyset line appears as a Line Appearance. The *GetKeysetPrimaryList* SOAP request is defined to accomplish this. Refer to Section 9.5, “Get Keyset Primary List”, on page 344 for more information.

4.13.3.1 Keyset Operation Mode

The Keyset Operation feature allows lines on a keyset device to be **DeviceBased** or **LineBased**. DeviceBased means that the data of the Keyset Primary Line is used for call origination and features that require that data and LineBased means that the data of this individual line is used for those purposes.

For each entry in the Line Appearance list, there are two associated fields that indicate the Originating and Terminating Keyset Operation Modes for that secondary line on that particular device (the device where the indexed subscriber is the Primary). This is because the Keyset Operation Modes can vary on a *per line, per device* basis. That is, a BGL that has an appearance on more than one device may be LineBased on one and DeviceBased on another, and operation modes may be different depending on whether the secondary line is the originating or terminating side of a call. If not specified, the default value is DeviceBased.

4.13.4 Call Forward LED Indication

This parameter is valid only for Keyset Primary Lines. It is used to indicate whether the line key LEDs on the device where this subscriber is the Keyset Primary will wink whenever those lines have call forwarding activated. This parameter is optional when creating a subscriber and when updating keyset info for a subscriber. If not supplied by the operator, the default value **True** is used.

Whenever the value of the Call Forward LED Indication attribute is changed by the operator, a function is called to send the `SRS_CFWD_INDICATION_CHG` event to the SIP Signaling Manager (SIP-SM), containing both the DN of the Primary Keyset subscriber being updated, and (if the device is registered) the IP address of the device.

4.14 Office Codes

Office Codes must be created before subscribers can be allocated to the system. Refer to Table 6 for office code field names, descriptions, and examples.

Field	Description	Example
OfficeCode	The digit sequence at the beginning of a range of directory numbers that represents the office in which these DNs are defined.	561981
AreaCodeLen	The length of the Area Code.	3
Charge Area	Mainly for Japan; indicates the charge information.	

Table 6 Office Code Fields

4.15 Home DNs

HomeDNs are the Directory Numbers that are available on the platform. HomeDN (Directory Numbers) must be created before subscribers can be allocated onto the system. Refer to Table 7 for DN fields, descriptions, and examples.

Field	Description	Example
OfficeCode	The office code of the home switch for this Directory Number.	561981
DirectoryNumber	The DN	

Table 7 HomeDN Fields

4.16 SIP Endpoints

The OpenScape Voice system supports SIP Endpoints in two varieties:

- with subscriber profile (SIP Subscribers)
- with endpoint profile

A SIP Endpoint with Subscriber Profile (automatically created when the Subscriber is created and automatically deleted when the subscriber is deleted) is referred to as a SIP Subscriber.

A SIP Endpoint with Endpoint Profile (explicitly created as a SIP Endpoint via SOAP or OpenScape Voice Assistant) is referred to as a SIP Endpoint. Refer to Section 4.17, “EndPoint Profiles”, on page 58 for more information.

4.17 EndPoint Profiles

Endpoint Profiles are used for SIP Network Servers such as SIP Gateways (access to the PSTN, HiPath 4000, HiPath 3000, RG 8700), SIP Proxy Servers (Comdasys), SIP Voice Mail Server (Xpressions), and SIP Conference Bridge. These profiles allow services and feature settings to be defined for these endpoints that need specific services. For example, some network servers require the OpenScape Voice system to deliver Calling ID information.

Endpoint Profiles may be associated with a Business Group, and thus with a Numbering Plan that is associated with the Business Group.

Endpoints are defined via the OpenScape Voice Assistant or CLI and can be defined via different authentication methods, such as:

- Trusted (No_SVC) — indicates that the endpoint message is a trusted network element. The endpoint messages are authenticated as coming from that trusted IP address or alias. There are no services or call routing definitions associated with that endpoint.

- Use Endpoint Profile and any Services (AUTH_EP) — indicates that the endpoint message is authenticated using the via header or entity name as being an alias. In this case, there exists an endpoint profile that may define services and/or routing definitions such as rate area or class or service.

Only those defined as AUTH_EP will use and execute EPP services.

Refer to Chapter 16, “Endpoint Profiles” for information regarding managing and provisioning Endpoint Profiles for SIP Network Servers.

4.18 SIP Network Servers

The OpenScape Voice system supports SIP network servers. They are defined on the OpenScape Voice system using the OpenScape Voice Assistant but profiles and feature profiles are created using the Assistant or the SOAP interface.

Before creating the SIP network server (or proxy server) via the Assistant, create a profile using the SOAP interface defined in this document. These SIP network server profiles are called Endpoint Profiles (EPPs). The EPPs contain various CODEC, QOS, and carrier information, as well as certain service attributes that apply to the SIP network server. For example, some SIP network servers require the system to provide the callerId information on their behalf or to implement Toll (Code) Restrictions. These services may be controlled using the SOAP interface defined in this document.

4.19 Dynamic CAC Router Management

In order to support the management of dynamic CAC (Call Admission Control) routers via an external Network Management System application, the SOAP Server provides several operations specific to this functional area in a separate interface definition (WSDL) file. The supported interfaces provide the capability to:

- Query all the access routers which have a secondary link (i.e. these are the access routers which are assigned to CAC groups in the OpenScape Voice system). The response returns the router addresses together with the interface names for the primary links in a list.
- Update the status of the primary link associated to an access router (i.e. main access link) to LinkUp/LinkDown. [R2210]
- Retrieve the current status of the primary link associated to a specific access router OR to retrieve the status the primary links associated to all access routers.
- Reset all Routers' link status to “LinkUp”.

For further details on performing these Dynamic CAC Router operations via the SOAP/XML interface, refer to Chapter 18, “Dynamic CAC Router Management”.

5 External Interfaces

5.1 Interface (WSDL) Versions

The format of the version is <Release>.<Version>.<Internal Revision>.

Beginning with Version 2.0, the interface version has been extended to include the following four values: <Release>.<Version>.<Build>.<Internal Revision>.

The internal revision is for internal use only and should not be used as it may change without affecting the existing interface. Table 8 lists the OpenScape Voice Versions and the corresponding WSDL versions.

Version/Patch	Initial WSDL Version	Current WSDL Version
Version 4R0	V4.00.12.01	V4.00.12.01
Version 4R1	V4.01.02.01	V4.01.31.02
Version 5	V5.00.15.01	V5.00.46.03
Version 6	V6.00.15.01	V6.00.37.02
Version 7R0	V7.00.14.02	V7.00.14.03
Version 7R1	V7.01.04.01	V7.01.09.05

Table 8 Interface Versions

5.1.1 The Interface WSDL and Interface Release Control

The interface WSDL version can be passed to some of the methods so that only those fields that are supported by the requested WSDL are returned. This assures the backward compatibility for SOAP clients that do not support the latest WSDL version. The Interface WSDL that is passed should follow the WSDL version format described above.

The InterfaceWSDL parameter has also been added to the SOAP/XML header structure, so now it is possible to pass the parameter for ALL requests if desired. Note that for APIs that already have an "InterfaceWSDL" as part of their request structure, if both are passed (i.e. in the header and in the request), the value in the request takes precedence.

It is recommended that all SOAP clients that are concerned with strict compliance ALWAYS specify the WSDL version in the SOAP header. This guarantees that these clients never receive unexpected fields in SOAP responses.

The following is a sample request that showing the use of the InterfaceWSDL in the header:

```
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/
envelope/"
xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/
encoding/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:unsp="urn:siemens-hiq8000"
SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/
soap/encoding/">
<SOAP-ENV:Header>
<unsp:hiqHEADER>
<InterfaceWSDL>@InterfaceWSDL@</InterfaceWSDL>
</unsp:hiqHEADER>
</SOAP-ENV:Header>
<SOAP-ENV:Body>
<unsp:GetFeatureList2>
<GetFeatureListRequest>
<ServiceFeatureType></ServiceFeatureType>
</GetFeatureListRequest>
</unsp:GetFeatureList2>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Table 9 lists the rules that control the Interface WSDL support level.

Release/Patch	Interface Release	WSDL for Support Check
Passed	N/A	Apply the passed Interface WSDL.
NIL	Passed	Apply the latest WSDL of the release passed.
NIL	NIL	Apply the latest WSDL of the latest release.

Table 9 Interface WSDL and Release Control Rules

5.2 Obsolete Method Names

All interfaces that accept multiple inputs are being changed to allow one input parameter and one output parameter to conform to newer industry standards.

The OpenScape Voice system supports the current and the previous two (2) release levels of the interface specification. This is the last release that supports the following *obsolete* methods. After this release, they will no longer be supported.

Old Method	New Method	Comments
AddBGMainNumber	AddBGMainNumber2	Industry standard
AddContactList	AddContactList2	Industry standard
AddSubToMlhg	AddSubToMlhg2	Industry standard
CancelJob	CancelJob2	Industry standard

Table 10 Required Interface Support (Sheet 1 of 4)

Old Method	New Method	Comments
CreateBG	CreateBG2	Industry standard
CreateBGDept	CreateBGDept2	Industry standard
CreateBgSubnet	CreateBgSubnet2	Industry standard
CreateDestCode	CreateDestCode2	Industry standard
CreateEpProfile	CreateEpProfile2	Industry standard
CreateFeatureProfile	CreateFeatureProfile2	Industry standard
CreateMlhg	CreateMlhg2	Industry standard
CreateMlhgTerm	CreateMlhgTerm2	Industry standard
CreateNumberPlan	CreateNumberPlan2	Industry standard
CreatePnpExtension	CreatePnpExtension2	Industry standard
CreatePnpLocationCode	CreatePnpLocationCode2	Industry standard
CreatePrefixAccessCode	CreatePrefixAccessCode2	Industry standard
CreateQOSProfile	CreateQOSProfile2	Industry standard
CreateSubscriber	CreateSubscriber2	Industry standard
DeleteBG	DeleteBG2	Industry standard
DeleteBGCpu	DeleteBGCpu2	Industry standard
DeleteBGDept	DeleteBGDept2	Industry standard
DeleteBGMainNumber	DeleteBGMainNumber2	Industry standard
DeleteBgSubnet	DeleteBgSubnet2	Industry standard
DeleteContactList	DeleteContactList2	Industry standard
DeleteDestCode	DeleteDestCode2	Industry standard
DeleteEpProfile	DeleteEpProfile2	Industry standard
DeleteFeatureProfile	DeleteFeatureProfile2	Industry standard
DeleteMlhg	DeleteMlhg2	Industry standard
DeleteMlhgTerm	DeleteMlhgTerm2	Industry standard
DeleteNumberPlan	DeleteNumberPlan2	Industry standard
DeletePnpExtension	DeletePnpExtension2	Industry standard
DeletePnpLocationCode	DeletePnpLocationCode2	Industry standard
DeletePrefixAccessCode	DeletePrefixAccessCode2	Industry standard
DeleteQOSProfile	DeleteQOSProfile2	Industry standard
DeleteSubscriber	DeleteSubscriber2	Industry standard
DisconnectSubscriber	DisconnectSubscriber2	Industry standard
GetBGAttendantNumbers	GetBGAttendantNumbers2	Industry standard
GetBGInfo	GetBGInfo2	Industry standard
GetBGInfoByOptions	GetBGInfoByOptions2	Industry standard
GetDestinationList	GetDestinationList2	Industry standard
GetEpInfo	GetEpInfo2	Industry standard

Table 10 Required Interface Support (Sheet 2 of 4)

Old Method	New Method	Comments
GetFeatureList	GetFeatureList2	Industry standard
GetJobInfo	GetJobInfo2	Industry standard
GetJobListByOption	GetJobListByOption2	Industry standard
GetMlhgInfo	GetMlhgInfo2	Industry standard
GetNumberPlanList	GetNumberPlanList2	Industry standard
GetPnPExtensionList	GetPnPExtensionList2	Industry standard
GetPnPLocationCodeList	GetPnPLocationCodeList2	Industry standard
GetPrefixAccessCodeList	GetPrefixAccessCodeList2	Industry standard
GetPrefixAccessCodeOne	GetPrefixAccessCodeOne2	Industry standard
GetSubscriberInfo	GetSubInfoByRel	Release independent i/f
GetSubscriberInfo2	GetSubInfoByRel	Release independent i/f
GetSubscriberInfoByOption	GetSubInfoByRel	Release independent i/f
GetSubscriberInfoByOption2	GetSubInfoByRel	Release independent i/f
HotDeskingLogOff	HotDeskingLogOff2	Industry standard
ModifyBGDept	ModifyBGDept2	Industry standard
ModifyDestCode	ModifyDestCode2	Industry standard
ModifyMlhgInfo	ModifyMlhgInfo2	Industry standard
ModifyMlhgTermInfo	ModifyMlhgTermInfo2	Industry standard
ModifyPnpExtension	ModifyPnpExtension2	Industry standard
ModifyPnpLocationCode	ModifyPnpLocationCode2	Industry standard
ModifyPrefixAccessCode	ModifyPrefixAccessCode2	Industry standard
ModifyQOSProfile	ModifyQOSProfile2	Industry standard
UpdateBGFeatures	UpdateBGFeatures2	Industry standard
UpdateBGMainNumber	UpdateBGMainNumber2	Industry standard
UpdateBGParms	UpdateBGParms2	Industry standard
UpdateBGStatus	UpdateBGStatus2	Industry standard
UpdateBgSubnet	UpdateBgSubnet2	Industry standard
UpdateConnectionInfo	UpdateConnectionInfo2	Industry standard
UpdateEpCapabilities	UpdateEpCapabilities2	Industry standard
UpdateEpFeatures	UpdateEpFeatures2	Industry standard
UpdateEpMgtInfo	UpdateEpMgtInfo2	Industry standard
UpdateEpPics	UpdateEpPics2	Industry standard
UpdateEpQos	UpdateEpQos2	Industry standard
UpdateFeatureProfile	UpdateFeatureProfile2	Industry standard
UpdateKeysetInfo	UpdateKeysetInfo2	Industry standard
UpdateSubscriberAccountMgtInfo	UpdateSubscriberAccountMgtInfo2	Industry standard

Table 10 Required Interface Support (Sheet 3 of 4)

Old Method	New Method	Comments
UpdateSubscriberAccountUserInfo	UpdateSubscriberAccountUserInfo2	Industry standard
UpdateSubscriberBGInfo	UpdateSubscriberBGInfo2	Industry standard
UpdateSubscriberCapabilities	UpdateSubscriberCapabilities2	Industry standard
UpdateSubscriberDn	UpdateSubscriberDn2	Industry standard
UpdateSubscriberFeatureProfile	UpdateSubscriberFeatureProfile2	Industry standard
UpdateSubscriberFeatures	UpdateSubscriberFeatures2	Industry standard
UpdateSubscriberPICs	UpdateSubscriberPICs2	Industry standard
UpdateSubscriberQOS	UpdateSubscriberQOS2	Industry standard
UpdateSubscriberStatus	UpdateSubscriberStatus2	Industry standard

Table 10 Required Interface Support (Sheet 4 of 4)

5.3 SOAP / XML Conventions

The OpenScape Voice system supports WSDL 1.1, SOAP 1.1 and SOAP 1.2 over HTTP 1.0, supporting Doc/literal and RPC encoding.

The OpenScape Voice system makes use of a couple of exceptions to standard XML that the client needs to be aware of:

- <Tag nil=true/> indicates that the tag has not been passed and is treated as if the tag is not passed
- <Tag/> indicates same as nil=true
- <Tag></Tag> is treated as an empty tag. If the tag data type is int or enum, the value passed is assumed to be 0. If the datatype if a string, the value passed is considered to be the empty string.

5.4 Primitive Data Definitions

The following fields are common in many methods.

5.4.1 Field Data Type

- **Numeric / Number** - a numeric field
- **Enum** - A field that accepts a value from an enumerated list. Although XML allows the client to specify the corresponding numeric value to be passed instead of the enumerated text, passing of the numeric is discouraged.
- **Character String** - a string of any characters. Some fields have restrictions on the character set allowed. A validity check that compares any entered string parameters against the RTP parameter Srx/Main/InvalidStringChars is performed. If a parameter contains any characters that are in this string, the input request will be rejected with an appropriate error response.
- There is a general restriction that the comma (,) may not be included as a part of any string field due to the current implementation of Subscriber Mass Provisioning (soapMassProv), which uses commas as delimiters. The same applies to quote marks (“). Some fields do allow commas and/or quote marks, those fields indicate so in their descriptions.
- **Numeric Character String** - a character strings that accepts only numerics. This field type allows leading Zeros.
- Note that in the Default Value column, the following definitions apply:
  - **N/A** (Not Applicable): This means that the concept of a default value does not apply in this context. For example, if the field is not entered and is not required, it simply doesn’t apply for the request, and thus there is no default. Also, where output fields are listed, this means that there is no default.
  - **required**: This means that the field is mandatory in this context. Since some value must be provided in the request, there is no default.
  - **““**: The empty string, used for character string fields.
  - Any other entry in the Default Value column represents the actual default value that is used.

5.4.2 Field Definitions

Attention: To avoid redundancy, detailed information for the data populated in the Length, Range or Default Value fields in Table 11 are outlined in Section 5.4.1, "Field Data Type", on page 66.

Field Name	Description	Length	Range	Default Value
AuthCode	External network access Authorization code. Maximum 50000 for each BG.	2..14	Numbers only. Must not have leading zeros.	N/A
BGDeptName	The name of the department.	1..20	Character String.	N/A
BGName	The name of a BG.	1..30	Character String	N/A (residential subscriber)
Billing	Indicates the type of billing. FlatRate (monthly) or UsageSensitive (per use). If not specified, defaults to FlatRate.	enum	FlatRate, UsageSensitive	N/A
COS (Class Of Service)	The Classes Of Service are used in call routing. The subscriber **may** optionally belong to a class of service. Outgoing calls are routed using the ClassOfService. During routing, if a subscriber belongs to a class of service, those routes are given preference; otherwise, the switches default class of service (unassigned) is used. If the subscriber has no ClassOfService assigned, the subscriber can access default routes only. Subscribers with COS specified will access COS routes first (if defined for the call that they make), if none are available, the default COS routes will be used. Defined ClassesOfService are retrievable using the GetCOSList interface method. They are definable on the switch using either the OpenScape Voice Assistant or CLI.		Character String	N/A
Country Code Length	The length of the country code associated with the subscriber's Home DN	integer		0
Enabled	Indicates whether a feature is enabled or not.	Boolean	True, False	False
FeatureProfileName	The name of a feature profile.	1..40	Character String	N/A

Table 11 Primitive Data Definitions (Sheet 1 of 6)

Field Name	Description	Length	Range	Default Value
Forwarding Destination	A directory number to be used as a forwarding destination for any of the call forward services or other service features that include a called party destination. In general this is the same as Service Id but the forwarding destination may be a length of up to 30 characters instead of 15.	1..30	Character String	N/A
Interface Release	The version for which information needs to be retrieved. Note that the InterfaceWSDL, with its higher granularity, is now the preferred parameter to indicate what data should be returned. For further details, please refer to Section 5.1, “Interface (WSDL) Versions”, on page 61.	Enum	Latest, Release6, Release8, Release9, Release10, Release10_1, Release10_	Latest
InterfaceWSDL	The WSDL version of the request. This indicates that only data that was introduced in the specified WSDL version or later shall be returned in response to this request. For further details, please refer to Section 5.1, “Interface (WSDL) Versions”, on page 61.	Any valid WSDL version.	Character string	N/A
Language	The language associated with a subscriber or endpoint profile.	1..30	Character String.	System default
Local	Local presence flags to display or reset the locally overwritten feature data	Structure	Local structure consists of Boolean for each data field could be modified locally.	N/A

Table 11 Primitive Data Definitions (Sheet 2 of 6)

Field Name	Description	Length	Range	Default Value
locCode	A digit string of {0-9} of max 14 digits represents the location code. The locCode consists of L2, L1, and L0 codes. The user interfaces will have the user enter the level codes in separate digit strings. The SOAP Server will concatenate into one digit string and derive the locL0Len, locL1Len, and locL2Len. L1 and L2 are optional. L1 cannot be administered without L0. L2 cannot be administered without L1. locL0Len - L0 code length, an integer {1-4}. Mandatory; defaults to 1. locL1Len - L1 code length, an integer {0-6}. Mandatory; defaults to 0. locL2Len - L2 code length, an integer {0-4}. Mandatory; defaults to 0. locSkipDigits - An integer (>=0). Defaults to 0. It indicates the number of leading digits in the location code to skip when converting the fully qualified PNP number to the extension.	1..14	See provisioned HomeDns	N/A
NumberPlan Name	The name of private Numbering Plan.	1..31	Character String	N/A
Operator ID	The Operator ID field is not used as part of call processing. It is used only for auditing and aiding the telco in identifying the subscriber. This field may be up to 20 characters in length. It is required in most interfaces.	1..40	Character String	N/A
Rate Area	The subscriber **may** optionally belong to a specific Rate Area. The rate area defines a local call verses a long distance call, as well as the routing for outbound calls. Outgoing calls are routed using the RateArea. During routing, if a subscriber belongs to a Rate Area, those routes are given preference; otherwise, the switch's default class of service (unassigned) is used. If the subscriber has no Rate Area assigned, the subscriber can access default routes only.The available list of Rate Areas can be retrieved via the GetRateAreaList interface method. They can be defined using either the OpenScape Voice Assistant or CLI.			N/A

Table 11 Primitive Data Definitions (Sheet 3 of 6)

Field Name	Description	Length	Range	Default Value
Service Id	The Directory Number for the subscriber. It is a fully qualified national number (NSN or National Significant Number). It does not include any spaces, parentheses, dashes, or access numbers. Example: 5619817000.	1..15	Character String	N/A
ServiceIdType	The OpenScape Voice system supports a number of different types of subscriber Dns.		PrimaryDn – identifies residential or BG subscribers, where the subscriber owns the connection. NonDnMlhgTerm – Identifies a MLHG terminal that does not have any DN, public or private. DisconnectDn - A DN that has been disconnected; it's subscriber data has been removed and only it's Home DN information remains.	PrimaryDn
Service FeatureType	The OpenScape Voice system supports different feature sets for different types of objects. The following are the types of objects that the OpenScape Voice system supports and that features are assignable to:		PrimaryDn – normal subscriber. BusinessGroup – a grouping of subscribers. Endpoint – SIP network servers such as SIP gateways. SvcFeatureProfileResidential - identifies features that apply to Residential Feature Profiles SvcFeatureProfileCentrex - identifies features that apply to Business Group Feature Profiles MlhgPIlot - identifies features that apply to MultiLine Hunt Group Pilot DNs.	PrimaryDn
			SIP - identifies features that apply to subscribers with connection type SIP.	

Table 11 Primitive Data Definitions (Sheet 4 of 6)

Field Name	Description	Length	Range	Default Value
Subscribed	Indicates the action to be taken for the service feature.		**Ignore** – skip this service/no action **AddReplace** – Add this feature and/or replace any existing definition of the feature that the subscriber might have. When a feature is AddReplace, the complete feature-specific data must be specified; missing fields will default as described in this document. **Modify** – The feature must already exist from either being previously provisioned at the Feature Profile Level (and inherited) or provisioned directly. Only the fields listed are modified. The presence fields may be sent to reset overrides for fields. **Delete** – Remove the feature from the subscriber. Inherited features may not be removed, although they may be denied.	Ignore
Subscribed, *continued*			**Denied** – Deny this feature to the subscriber. Do not allow the subscriber to inherit the feature from a Feature Profile. **Inherit** – This is a read-only setting returned in the GetSubscriber Info interface variants. This value indicates that the subscriber has the feature from the subscriber's FP. **SwitchWide** – This is a read-only setting returned in the GetSubscriber Info interface variants for a small subset of available features. This value indicates that the feature is set switch wide and thus all subscribers have access to it.	

Table 11 Primitive Data Definitions (Sheet 5 of 6)

Field Name	Description	Length	Range	Default Value
TOY	Date and Time fields are in xsd__datTime format. Whenever a time of day applies without a date, the date portion is specified as 0000-00-00. Timezone offsets are not supported. Seconds will only be transmitted in the ResultTOY field and not on any subscriber data.	20	CCYY-MM-DDThh:mm	N/A
Voip. Bearer_ Speech	Indicates whether it supports this capability.	Boolean	True, False	True
Voip. Bearer_3_1K	Indicates whether it supports this capability.	Boolean	True, False	False
Voip. Bearer_56K	Indicates whether it supports this capability.	Boolean	True, False	False
Voip. Bearer_64K	Indicates whether it supports this capability.	Boolean	True, False	False
Voip. FaxMode	Indicates the supported fax modes.	enum	None, G711, T38, T38withG711FallBack	G711
Voip. Echo Cancellation	Indicates whether it supports this capability.	Boolean	True, False	False
Voip. Silence Suppression	Indicates whether it supports this capability.	Boolean	True, False	False
Voip.Fax Statistics	Indicates whether it supports this capability.	Boolean	True, False	False
Voip.Outband DTMF Signaling	Indicates whether it supports this capability.	Boolean	True, False	False
Voip. Suppress OutbandDTMFWith G711	Indicates whether to suppress Outband DTMF if G711 codec is in use.	Boolean	True, False	False

Table 11 Primitive Data Definitions (Sheet 6 of 6)

5.5 Paging

For the SOAP requests that accomplish Get List functionality, it is desirable that a paging mechanism be used to facilitate paging through the result set of the returned data by SOAP clients, such as OpenScape Voice Assistant that display the data graphically to the user. Therefore, the following paging mechanism exists to be used by the various Get List SOAP requests, as necessary.

For Get List SOAP requests, a paging structure is defined in the WSDL interface that includes the following fields:

- Page Size — the number of rows to be shown on a page. This is the number of rows displayed in the GUI. Must be >= 1 and the maximum of 1000.
- Page Number — The page that will be shown in the GUI. Must be >= 1 or be equal to -10, which indicates the last page.

Depending on the individual Get List request, there may be additional input parameters, including filter criteria, to apply to the existing objects on the system.

For Get List SOAP requests, a PaginatedInfo structure is defined in the WSDL interface that includes the following fields:

- Number of rows in the result set (total number of entries meeting the filter criteria, if any)
- Number of pages in the result set (total number of pages of data, given the number of rows and the Page Size)
- Page Size (same as Page Size in request)
- Page Number (same as Page Number in request)

Note that paging is optional in order to make GetList SOAP requests backward compatible. If no paging information is sent, the entire list, up to the maximum possible elements, is returned. The following default behavior applies to paging requests:

- If PageSize is <= 0 or >= the total number of existing elements, no paging is assumed and the entire list is returned.
- If PageNumber == -10, or >= the last page, the last page is displayed (with the starting entry as it is specified with the last page number, so the size of this page may be less than the specified PageSize)
- If PageNumber <= 0, the first page is displayed.

The PaginatedInfo structure in the response will reflect the CurrentPage info according to the default behavior, not according to the actual user input for Paging.

Note: Sorting is a separate issue from Paging and is not addressed in this document.

5.6 Return Code Information

All request responses contain return code information. These fields are described in Section 25.1, “Result Code Information”, on page 878.

6 Security on OpenScape Voice and the OS Voice SOAP Server Interface

OpenScape Voice and the OpenScape Voice SOAP server provide the following types of security:

- Client Authorization

  The scope of these APIs is to provide authentication and authorization when receiving XML requests from a SOAP client. This will result in a secure TLS functionality along with a controlled access to all SOAP APIs to provide insurance that clients cannot perform any operation on the switch unless they are authorized to do so.

- Authentication

  The SOAP server shall enable TLS. The secured SOAP server using TLS will run on starting port 8757, creating new instances. Both starting port and number of instances are defined by RTPparameters (Srx/Subp/StartingPortWithTLS and Srx/Subp/NumberOfInstancesWithTLS). This means that on the switch there will be secured (8757 - (8757 + number of instances)) and non-secured (8767 - (8767 + number of instances)) ports.

- Authorization

  There will be a new RTP parameter defined on the switch (Srx/Subp/Authorization) which will define whether authorization is required. For backwards compatibility, if this parameter is set to false, all clients should have access to all APIs. If it set to true, this means that the customer wants to authorize the SOAP clients on the switch and all clients will be checked before performing any operation through the SOAP APIs.

  When authorization is requested there will be one IP address that will belong to the super administrator. The super administrator will have the authority to determine all the other clients' privileges on the SOAP requests. This will be implemented via the client profile concept. The super administrator will create a client profile for each client and determine if this is a secured client or not, his role (or roles if multiple), which can be SuperAdmin, SystemAdmin, Residential, Network Admin, BG admin, CAC Admin, or LAES Admin, the access state (Normal or Blocked) to the SOAP APIs, and the lists of BGs and CAC groups that the client can access. If authorization is required, ALL clients should have a client profile.

6.1 Use of TLS with the OSV Soap Server

6.1.1 Configuration of RTP Parameters

The TLS functionality for the OSV SOAP/XML is enabled through the following RTP parameters:

- Srx/Subp/StartingPortWithTLS default value: 8757
- Srx/Subp/NumberOfInstancesWithTLS default value: 1

By default the OSV SOAP/XML interface spawns one child process that accepts TLS connections at port 8757. Up to 4 consecutive ports can be opened, with variable offset.

The OSV SOAP/XML interface requires **mutual authentication** in SSL handshaking. The certificate used for each of the (maximum 4) children processes that can be spawned, may differ and it's the one to be found in the path/filename specified by the following RTP parameters:

RTP Parameter	Description	Default Value
`SSL/Soap/Server/KeysPath`	The **PATH** of the directory where the certificate that the OSV SOAP Server would use in TLS connections, resides.	`/usr/local/ssl/private`
`SSL/Soap/Server/Keys1`	The **FILENAME** of the certificate that the SOAP Server would use in TLS connections on port [Srx/Subp/StartingPortWithTLS] . This certificate needs to be in PEM format.	`server.pem`
`SSL/Soap/Server/Keys2`	The **FILENAME** of the certificate that the SOAP Server would use in TLS connections on port [Srx/Subp/StartingPortWithTLS]+1 . This certificate needs to be in PEM format.	`server.pem`
`SSL/Soap/Server/Keys3`	The **FILENAME** of the certificate that the SOAP Server would use in TLS connections on port [Srx/Subp/StartingPortWithTLS]+2 . This certificate needs to be in PEM format.	`server.pem`
`SSL/Soap/Server/Keys4`	The **FILENAME** of the certificate that the SOAP Server would use in TLS connections on port [Srx/Subp/StartingPortWithTLS]+3 . This certificate needs to be in PEM format.	`server.pem`

Table 12 RTP Parameters for SOAP Server Keys

The certificate(s) above need to have been created and signed by the CA authority whose signature is stored in the pem file under the path/filename specified by the following RTP parameters:

RTP Parameter	Description	Default Value
`SSL/Soap/Server/CertificatesPath`	The PATH of the directory where certificates from Certificate Authorities are held for SOAP Servers. Please check the man page for [http:www.openssl.org/docs/ssl/SSL_CTX_load_verify_locations.html SSL_CTX_load_verify_locations] on the format of certificates and how this parameter is used in conjunction with the parameter *SSL/Soap/Server/SupportedCertificates*.	`/usr/local/ssl/certs`
`SSL/Soap/Server/Supported-Certificates`	The FILENAME of the file that contains the list of certificates from Certificate Authorities that are held for SOAP Servers. Please check the man page for [http:www.openssl.org/docs/ssl/SSL_CTX_load_verify_locations.html SSL_CTX_load_verify_locations] on the format of certificates and how this parameter is used in conjunction with the parameter *SSL/Soap/Server/CertificatesPath*. If this file is changed, then TTUD needs to be restarted to re-read the file.	`root.pem`

Table 13 RTP Parameters for SOAP Server Certificates

6.1.2 Certificates and TLS Verification

A brief example of using the OSV *soapServer* and onboard test SOAP client (*TestSendSoapReq)* to check TLS handshaking is described below:

1. Create a custom root certificate:
   - `openssl req -newkey rsa:1024 -sha1 -keyout rootkey.pem -out rootreq.pem -config root.cnf`
   - `openssl x509 -req -in rootreq.pem -sha1 -extfile root.cnf -extensions v3_ca -signkey rootkey.pem -out rootcert.pem -days 3650`
   - `cp rootcert.pem root.pem`
   - `mkdir soap_certs`

- `mv root.pem ./soap_certs`
- Install the certificate from the soap_certs directory: `/usr/bin/c_rehash $PWD`

2. Create a custom server certificate:
   - `openssl req -newkey rsa:1024 -sha1 -keyout serverkey.pem -out serverreq.pem -config server.cnf -reqexts v3_req -nodes`
   - `openssl x509 -req -in serverreq.pem -sha1 -extfile server.cnf -extensions v3_req -CA rootcert.pem -CAkey rootkey.pem -CAcreateserial -out servercert.pem -days 3650cat servercert.pem`
   - `serverkey.pem rootcert.pem > server.pem`

3. **Create a custom client certificate (**the client in this case will be onboard OSV, i.e. its test tool *TestSendSoapReq)* **:**
   - `openssl req -newkey rsa:1024 -sha1 -keyout clientkey.pem -out clientreq.pem -config client.cnf –reqexts v3_req –nodes`
   - `openssl x509 -req -in clientreq.pem -sha1 -extfile client.cnf -extensions v3_req -CA rootcert.pem -CAkey rootkey.pem -CAcreateserial -out clientcert.pem -days 3650`
   - `cat clientcert.pem clientkey.pem rootcert.pem > client.pem`
   - Move the client.pem file inside `/usr/local/ssl/private` which is the default directory for storing the client files

4. Create a sample SOAP envelope (to be sent by the test-tool *TestSendSoapReq SOAP client, to the OSV soapServer)* and store it on a text file (e.g. inside `/unisphere/srx3000/srx/AAA.xml`) with the following content::

```
<TEST SSL_HANDSHAKING>

<?xml version="1.0" encoding="UTF-8"?>

<SOAP-ENV:Envelope

 xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
```

```
 xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/
encoding/"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance"
 xmlns:xsd="http://www.w3.org/2001/XMLSchema"
 xmlns:unsp="urn:siemens-hiq8000">
 <SOAP-ENV:Body encodingStyle="http://
schemas.xmlsoap.org/soap/encoding/">
  <unsp:GetLicensingInfo>
   <GetLicensingInfoRequest>
    <OperatorID>username</OperatorID>
    <LicensingType>EndUserLicense</LicensingType>
   </GetLicensingInfoRequest>
  </unsp:GetLicensingInfo>
 </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

</TEST SSL_HANDSHAKING>
```

5. Create two new folders under `/usr/local/ssl`

   One will be used for storing the root.pem certificate: `/usr/local/ssl/soap_certs`

   The other will be used for storing the server.pem certificate: `/usr/local/ssl/soap_private`

   Usie Rtp Admin Cli to change the values of the following Rtp Parameters

   `SSL/Soap/Server/CertificatesPath to` /usr/local/ssl/soap_certs

   `SSL/Soap/Server/KeysPath to`/ usr/local/ssl/soap_private

6. Execute the following command on an OSV console to spawn a private SSL OSV SOAP child with SSL support on port 8758:

   ```
   $/unisphere/srx3000/UNSPsubp/soapServer -d -p=8758 -
   child -ssl
   ```

7. Execute the following command on an OSV console to send a SOAP envelope via the test SOAP client *TestSendSoapReq* to the OSV soapServer child spawned in the previous step.

   ```
   $/unisphere/srx3000/UNSPsubp/TestSendSoapReq -p=8758
   -f=../srx/AAA.xml -format -ssl
   ```

A response that is similar to the following example is expected:

```
Host name is <grd424vm1n1> IP address is <10.4.242.10>
SSL connection using AES256-SHA
Server certificate:
        subject: /C=GR/ST=ATHENS/L=KIFISIA/O=SIEMENS/
OU=SOAP/CN=DIMITRIS
         issuer: /C=GR/ST=ATHENS/L=ATHENS/O=KIFISIA/
OU=SOAP/CN=DIMITRIS

HTTP/1.1 200 OK
Server: gSOAP
<ResultCode>0
<ResultText1>Success
<ResultText2>
<ResultText3>
<ResultText4>
<?xml>

  <SOAP-ENV:Envelope>
    <SOAP-ENV:Body>
      <unsp:GetLicensingInfoResult>
        <Result>
          <ResultCode>0</ResultCode>
          <ResultMsgNumber>0</ResultMsgNumber>
          <ResultTOY>2009-07-02T16:11:32</ResultTOY>
          <ResultText1>Success</ResultText1>
          <ResultText2></ResultText2>
          <ResultText3></ResultText3>
          <ResultText4></ResultText4>
        </Result>
        <LicensingInfo>
         <LicensingType>EndUserLicense</LicensingType>
          <InstalledCount>100000</InstalledCount>
          <UsedCount>11</UsedCount>
        </LicensingInfo>
      </unsp:GetLicensingInfoResult>
    </SOAP-ENV:Body>
  </SOAP-ENV:Envelope>
```

8. If we log in on another OSV and repeat the process above without having installed the proper client certificates and send the request to the previous OSV SOAP host/port (given that the port is not blocked by PFRs), the request should be rejected (note that the command below now contains the IP of the host of the SSL enabled OSV soapServer:

```
$/unisphere/srx3000/UNSPsubp/TestSendSoapReq -
h=10.4.242.10 -p=8757 -f=../srx/AAA.xml -format –ssl
```

A response that is similar to the following example is expected:

```
SSL Verify error with certificate at depth 1: self
signed certificate in certificate chain
```

```
certificate issuer /C=GR/ST=ATHENS/L=ATHENS/
O=KIFISIA/OU=SOAP/CN=DIMITRIS
certificate subject /C=GR/ST=ATHENS/L=ATHENS/
O=KIFISIA/OU=SOAP/CN=DIMITRIS
SSL connection using AES256-SHA
Server certificate:
        subject: /C=GR/ST=ATHENS/L=KIFISIA/O=SIEMENS/
OU=SOAP/CN=DIMITRIS
         issuer: /C=GR/ST=ATHENS/L=ATHENS/O=KIFISIA/
OU=SOAP/CN=DIMITRIS
```

6.2 Create Client Profile

This interface creates a new client profile on OpenScape Voice.

Table 14 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes	
0		subRC_SUCCESS	
		subRC_APP_CP_NAME_LEN_INVALID	32200001 The client profile's name has invalid length
		subRC_DB_CP_PROFILE_LOOKUP_FAILED	12200002 Failed to find client profile in the database
		subRC_APP_CP_PROFILE_EXIST	32200005 Client profile doesn't exist in the database
		subRC_APP_FEAT_CPLIST_ENTRY_MAX_NUMBER_EXCEED	32200006 The entries in the client profile's list have exceeded the maximum number allowed.
		subRC_APP_NUM_CLIENT_PROFILES_OUT_OF_RANGE	32200007 The number of client profiles have exceeded the maximum number allowed.
		subRC_APP_CP_INSERTION_FAILED	32200008 Failed to insert the client profile.
		subRC_APP_FEAT_CPLIST_ENTRY_DATA_INVALID	32200010 Entry data is invalid.

Table 14 Create Client Profile Return Codes (Sheet 1 of 2)

WSDL Version	Obsolete in WSDL Version	Return Codes	
		subRC_APP_CP_PROFILE_ROLE_NOT_SPECIFIED	32300015 The Role of the Client is missing
		subRC_APP_CP_BGLIST_ENTRY_DATA_MISSING	32300016 BG List's entry not passed but required
		subRC_APP_CP_CACLIST_ENTRY_DATA_MISSING CAC	32300017 List's entry not passed but required
		subRC_APP_CP_PROFILE_DATA_MISSING	32300020 Client's Profile Information missing

Table 14 Create Client Profile Return Codes (Sheet 2 of 2)

Table 15 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default
operatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. Appendix			
AllOrNone	A flag to indicate whether atomic processing is required or partial success is allowed. If set to true, when the processing of one entry fails, the data is rolled back to its state prior to the request. If set to false, when the processing of one entry fails, a best effort attempt will be made, and the subsequent entries will be processed.	Boolean	True/false	false
ClientId	This can be an IP address of the client or the name (Owner) of the client.	16	Character string	
ClientStatus	The status of the client.	enum	CS_Blocked CS_Active	
isSecuredClient	Indicates whether the client runs in TLS mode so it will adhere to the TLS SOAP server.	Boolean	true/false	false

Table 15 Create Client Profile Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default
ClientRole	The role that the client has on the switch.	enum	CR_SuperAdmin CR_SystemAdmin CR_ResidentialAdmin CR_NetworkAdmin CR_BusinessGroupAdmin CR_CACAdmin CR_CaleaAdmin CR_ExecutiveAssistantAdmin	required
BusinessGroupList	The list of all business groups that the client can access.	List	Each entry of the list is a character string representing the BG name. Please refer to Please refer to Section 5.4, “Primitive Data Definitions”, on page 66 for data type definitions.	N/A
CACGroupList	The list of all CAC groups that the client can access.	List	Each entry of the list is a character string representing the CAC group name (max length 31).	N/A

Table 15 Create Client Profile Input Fields (Sheet 2 of 2)

Table 16 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
Result	The result of the operation.		Type unsp__ResultCodeStruct	

Table 16 Create Client Profile Output Fields

6.3 Modify Client Profile

This interface modifies an existing client profile.

Table 17 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes	
0		subRC_SUCCESS	
		subRC_APP_CP_DOES_NOT_EXIST	32200000 The client profile does not exist.
		subRC_APP_CP_NAME_LEN_INVALID	32200001 The client profile's name has invalid length

Table 17 Modify Client Profile Return Codes (Sheet 1 of 2)

WSDL Version	Obsolete in WSDL Version	Return Codes	
		subRC_DB_CP_PROFILE_LOOKUP_FAILED	12200002 Failed to find client profile in the database
		subRC_APP_FEAT_CPLIST_ENTRY_MAX_NUMBER_EXCEED	32200006 The entries in the client profile's list have exceeded the maximum number allowed.
		subRC_APP_FEAT_CPLIST_ENTRY_DATA_INVALID	32200010 Entry data is invalid.
		subRC_APP_CP_PROFILE_ROLE_NOT_SPECIFIED	32300015 The Role of the Client is missing
		subRC_APP_CP_BGLIST_ENTRY_DATA_MISSING	32300016 BG List's entry not passed but required
		subRC_APP_CP_CACLIST_ENTRY_DATA_MISSING CAC	32300017 List's entry not passed but required
		subRC_APP_CP_PROFILE_MODIFICATION_FAILED	32300018 Client Profile Data update failed
		subRC_APP_CP_PROFILE_DATA_MISSING	32300020 Client's Profile Information missing

Table 17 Modify Client Profile Return Codes (Sheet 2 of 2)

Table 18 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default
operatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
AllOrNone	A flag to indicate whether atomic processing is required or partial success is allowed. If set to true, when the processing of one entry fails, the data is rolled back to its state prior to the request. If set to false, when the processing of one entry fails, a best effort attempt will be made, and the subsequent entries will be processed.	Boolean	True/false	false

Table 18 Modify Client Profile Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default
ClientId	This can be an IP address of the client or the name (Owner) of the client.	16	Character string	
ClientStatus	The status of the client.	enum	CS_Blocked CS_Active	
isSecuredClient	Indicates whether the client runs in TLS mode so it will adhere to the TLS SOAP server.	Boolean	true/false	false
ClientRole	The role that the client has on the switch.	enum	CR_SuperAdmin CR_SystemAdmin CR_ResidentialAdmin CR_NetworkAdmin CR_BusinessGroupAdmin CR_CACAdmin CR_CaleaAdmin CR_ExecutiveAssistantAdmin	required
BusinessGroupList	The list of all business groups that the client can access.	List	Each entry of the list is a character string representing the BG name. Please refer to Section 5.4, “Primitive Data Definitions”, on page 66 for data type definitions.	N/A
CACGroupList	The list of all CAC groups that the client can access.	List	Each entry of the list is a character string representing the CAC group name (max length 31).	N/A

Table 18 Modify Client Profile Input Fields (Sheet 2 of 2)

6.4 Delete Client Profile

This interface removes a client profile from OpenScape Voice.

Table 19 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes	
0		subRC_SUCCESS	
		subRC_APP_CP_NAME_LEN_INVALID	32200001 The client profile's name has invalid length
		subRC_DB_CP_PROFILE_LOOKUP_FAILED	12200002 Failed to find client profile in the database

Table 19 Delete Client Profile Return Codes (Sheet 1 of 2)

WSDL Version	Obsolete in WSDL Version	Return Codes	
		subRC_DB_CP_DELETE_FAILURE	12200013 Deletion of the client profile has failed
		subRC_APP_CP_PROFILE_DATA_MISSING	32300020 Client's Profile Information missing

Table 19 Delete Client Profile Return Codes (Sheet 2 of 2)

Table 20 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default
operatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
operatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			

Table 20 Delete Client Profile Input Fields

6.5 Get Client Profile

This interface displays the client authorization information of a specific client.

Table 21 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes	
0		subRC_SUCCESS	
		subRC_APP_CP_NAME_LEN_INVALID	32200001 The client profile's name has invalid length
		subRC_APP_CP_PROFILE_EXIST	32200005 Client profile doesn't exist in the database
		subRC_APP_CP_BGLIST_ENTRY_DATA_MISSING	32300016 BG List's entry not passed but required
		subRC_APP_CP_CACLIST_ENTRY_DATA_MISSING	32300017 CAC List's entry not passed but required
		subRC_APP_CP_PROFILE_DATA_MISSING	32300020 Client's Profile Information missing

Table 21 Get Client Profile Return Codes

Table 22 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default
operatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
ClientId	This can be an IP address of the client or the name (Owner) of the client. Required	16	Character string	

Table 22 Get Client Profile Input Fields

6.6 Get Client Profile List

This interface returns all the clients that have access to OpenScape Voice.

Table 23 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes	
0		subRC_SUCCESS	
		subRC_DB_CP_PROFILE_LOOKUP_FAILED	12200002 Failed to find client profile in the database.
		subRC_SYS_CP_CONTACT_ALLOC_ERR	12200014 Allocation error.

Table 23 Get Client Profile List Return Codes

Table 24 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default
operatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
Paging	Paging structure for get list requests. For more information, refer to Section 5.5, “Paging”, on page 72.			

Table 24 Get Client Profile List Input Fields

6.7 Create PFR (Packet Filter Rule)

This interface creates a new packet filter rule on OpenScape Voice.

Note that ‘system’ PFR entries (i.e. prefixed with ‘System’) cannot be created they are pre-existing in the system.

Table 25 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
V7.00.16.01	N/A	subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_PFR_ACCESS_SYSTEM_ENTRY subRC_APP_PFR_NAME_LEN_INVALID subRC_APP_PFR_DESCRIPTION_LEN_INVALID subRC_APP_PFR_LOCAL_ALIAS_LEN_INVALID subRC_APP_PFR_REMOTE_FQDN_INVALID subRC_APP_PFR_REMOTE_IP_LEN_INVALID subRC_APP_PFR_REMOTE_NEMASK_LEN_INVALID subRC_APP_PFR_INVALID

Table 25 Create Packet Filter Return Codes

Table 26 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default
operatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
Name	The Packet Filter rules name.	1..63	string	required
Description	User supplied text.	1..63	string	N/A
Protocol	The protocol for this PFR	Enum	PROTO-ICMP PROTO-UDP PROTO-TCP PROTO-ALL PROTO-ESP PROTO-AH PROTO-SCTP PROTO-ICMPV6	PROTO-ALL
Direction	The direction for this PFR	Enum	DIR-INCOMING DIR-OUTGOING DIR-BOTH-WAYS	DIR-INCOMING
Action	Whether the rule permits or blocks the packets	Enum	• ACTION-ALLOW • ACTION-DROP	
LocalAlias	Alias	1..63	string	N/A
LocalPortBegin	Local IP port begin	N/A	int	0
LocalPortEnd	Local IP port end	N/A	int	0
LocalPortExpression	Local Port Expression		string	N/A
RemoteFQDN	Remote Fully Qualified Name	1..63	string	N/A
RemoteIP	Remote IP Address	1..50	string	N/A
RemoteNetmask	Remote IP Netmask	1..15	string	N/A
RemotePortBegin	Remote IP port begin	N/A	int	0

Table 26 Create Packet Filter Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default
RemotePortEnd	Remote IP port end	N/A	int	0

Table 26 Create Packet Filter Input Fields (Sheet 2 of 2)

Table 27 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
Result	The result of the operation.		Type unsp__ResultCodeStruct	

Table 27 Create Packet Filter Output Fields

6.8 Modify PFR (Packet Filter Rule)

This interface modifies a packet filter rule on OpenScape Voice.

Note that ‘system’ PFR entries (i.e. prefixed with ‘System’) cannot be created they are pre-existing in the system.

Table 28 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
V7.00.16.01	N/A	subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_PFR_ACCESS_SYSTEM_ENTRY subRC_APP_PFR_NAME_LEN_INVALID subRC_APP_PFR_DESCRIPTION_LEN_INVALID subRC_APP_PFR_LOCAL_ALIAS_LEN_INVALID subRC_APP_PFR_REMOTE_FQDN_INVALID subRC_APP_PFR_REMOTE_IP_LEN_INVALID subRC_APP_PFR_REMOTE_NEMASK_LEN_INVALID subRC_APP_PFR_INVALID

Table 28 Modify Packet Filter Return Codes

Table 29 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default
operatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
Name	The Packet Filter rules name.	1..63	string	required
Description	User supplied text.	1..63	string	N/A

Table 29 Modify Packet Filter Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default
Protocol	The protocol for this PFR	Enum	PROTO-ICMP PROTO-UDP PROTO-TCP PROTO-ALL PROTO-ESP PROTO-AH PROTO-SCTP PROTO-ICMPV6	PROTO-ALL
Direction	The direction for this PFR	Enum	DIR-INCOMING DIR-OUTGOING DIR-BOTH-WAYS	DIR-INCOMING
Action	Whether the rule permits or blocks the packets	Enum	• ACTION-ALLOW • ACTION-DROP	
LocalAlias	Alias	1..63	string	N/A
LocalPortBegin	Local IP port begin	N/A	int	0
LocalPortEnd	Local IP port end	N/A	int	0
RemoteFQDN	Remote Fully Qualified Name	1..63	string	N/A
RemoteIP	Remote IP Address	1..50	string	N/A
RemoteNetmask	Remote IP Netmask	1..15	string	N/A
RemotePortBegin	Remote IP port begin	N/A	int	0
RemotePortEnd	Remote IP port end	N/A	int	0

Table 29 Modify Packet Filter Input Fields (Sheet 2 of 2)

Table 30 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
Result	The result of the operation.		Type unsp__ResultCodeStruct	

Table 30 Modify Packet Filter Output Fields

6.9 Delete PFR (Packet Filter Rule)

This interface deletes a packet filter rule on OpenScape Voice.

Note that ‘system’ PFR entries (i.e. prefixed with System) cannot be deleted.

Table 31 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
V7.00.16.01	N/A	subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_PFR_ACCESS_SYSTEM_ENTRY subRC_APP_PFR_NAME_LEN_INVALID subRC_APP_PFR_INVALID

Table 31 Delete Packet Filter Return Codes

Table 32 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default
operatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
Name	The Packet Filter rules name.	1..63	string	required

Table 32 Delete Packet Filter Input Fields

Table 33 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
Result	The result of the operation.		Type unsp__ResultCodeStruct	

Table 33 Delete Packet Filter Output Fields

6.10 Query PFR (Packet Filter Rule)

This interface returns a list of information for packet filter rules on OpenScape Voice.

Notice that in contrast to Create/Modify/Delete PFR APIs ‘system’ PFR entries (i.e. prefixed with System) are queried and returned normally so that the administrator can view all PFR entries of the system.

Table 34 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
V7.00.16.01	N/A	subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_PFR_NAME_LEN_INVALID subRC_APP_PFR_DESCRIPTION_LEN_INVALID subRC_APP_PFR_LOCAL_ALIAS_LEN_INVALID subRC_APP_PFR_QUERY_ERROR

Table 34 Query Packet Filter Return Codes

Table 35 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
InterfaceWSDL			string	
Paging				
SortType	Sort Type	Enum	• SortBy-LocalAlias • SortBy-LocalPortBegin • SortBy-LocalPortEnd	SortBy-LocalAlias
SortOrder	Desired sort order	Enum	• SortOrderAscending • SortOrderDescending	
Name	The Packet Filter rule name.	1..63	string	
LocalAlias	Alias	1..63	string	
LocalPortBegin	Local IP port begin	N/A	int	
LocalPortEnd	Local IP port end	N/A	int	

Table 35 Query Packet Filter Input Fields

Table 36 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
Result	The result of the operation.		Type unsp__ResultCodeStruct	
PaginatedInfo				
QueryPFRList	The list of PFRs struct			
QueryPFRElemen	Elements in the list			
Name	The Packet Filter rules name.	1..63	string	
Description	User supplied text.	1..63	string	
Protocol	The protocol for this PFR	Enum	PROTO-ICMP PROTO-UDP PROTO-TCP PROTO-ALL PROTO-ESP PROTO-AH PROTO-SCTP PROTO-ICMPV6	
Direction	The direction for this PFR	Enum	DIR-INCOMING DIR-OUTGOING DIR-BOTH-WAYS	
Action	Whether the rule permits or blocks the packets	Enum	• ACTION-ALLOW • ACTION-DROP	
LocalAlias	Alias	1..63	string	
LocalPortBegin	An IP port	N/A	int	
LocalPortEnd	An IP port	N/A	int	
RemoteFQDN	Fully Qualified Name	1..63	string	

Table 36 Query Packet Filter Output Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
RemoteIP	IP Address	1..50	string	
RemoteNetmask	IP Netmask	1..15	string	
RemotePortBegin	An IP port	N/A	int	
RemotePortEnd	An IP port	N/A	int	

Table 36 Query Packet Filter Output Fields (Sheet 2 of 2)

6.11 Get PFR Symbolic Name List

This interface returns information for the PFR symbolic names on OpenScape Voice.

Table 37 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
V7.00.16.01	N/A	subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_SYMBOLIC_NAME_QUERY_ERROR

Table 37 Get PFR Symbolic Name Return Codes

Table 38 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
InterfaceWSDL			string	
AliasName			string	

Table 38 Get PFR Symbolic Name List Input Fields

Table 39 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
Result	The result of the operation.		Type unsp__ResultCodeStruct	
SymbolicNameList	Symbolic Name list structure			
SymbolicNameElem ent	Elements in the lists			
SymbolicName	Symbolic Name		string	
AliasElement1				
.Alias	Alias of Element 1		string	
.IPAddressElement1.	IP Address		string	

Table 39 Get PFR Symbolic Name List Output Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
.IPAddressElement2	IP Address		string	
AliasElement2				
.Alias	Alias of Element 2		string	
.IPAddressElement1	IP Address		string	
.IPAddressElement2	IP Address		string	

Table 39 Get PFR Symbolic Name List Output Fields (Sheet 2 of 2)

7 General Utility Interfaces

This is a collection of Application Programming Interfaces (APIs) about objects that are not necessarily provisioned by SOAP. Some of this information, however, may be needed while provisioning other objects.

7.1 Get Version

The GetVersion method returns build and release information such that the user of this interface may know what interface methods are available on the system. See Section 5.1, “Interface (WSDL) Versions”, on page 61 for a description of the WSDL numbering nomenclature. Also refer to Section 3.6, “Requirements for Using this Interface”, on page 32.

Table 40 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
1.0.6		subRC_SUCCESS

Table 40 Get Version Return Codes

Table 41 lists the *output* fields for this method.

Field Name	Description
Version	The WSDL version of the SOAP Server. For example, 10.00.08.01.
MarketBuild	The OpenScape Voice-build version. For example, 10.01.01.ALL.22.
SoftSwitch Environment	Enterprise or Carrier.
buildDateTime	The time the SOAP Server process was built. For example, 2006-01-18T09:34.

Table 41 Get Version Output Fields

7.2 Get Feature List

The GetFeatureList method is used to obtain the list of features enabled on the system for the specified category. If no category is specified, the residential feature list is returned.

Table 42 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
1.0.5		subRC_SUCCESS

Table 42 Get Feature List Return codes

Table 43 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default Value
ServiceFeatureType	The category of Features to retrieve.	enum	PrimaryDn_, BusinessGroup_, Endpoint_, BusinessGroupLine_, SvcFeatureProfileResidential, SvcFeatureProfileCentrex	PrimaryDn_

Table 43 Get Feature List Input Fields

Table 44 lists the *output* fields for this method.

Field Name	Description
ServiceFeatureType	Feature Category as explained in the Input Field Description. E.g “PrimaryDn”
GetServiceFeatureList Data (List)	FeatureId - The internal feature Id. For example, “CfVM”.
	FeatureNumber – The feature number For example, “99”.
	FeatureDesc – Textual description of the Feature. For example, Voice Mail.

Table 44 Get Feature List Output Fields

7.3 Get Config Data

This API returns a list of all OpenScape Voice (RTP) configuration parameters relevant to subscriber provisioning and their current values. If a parameter could not be retrieved from RTP, the default value used by subscriber provisioning is returned.

Table 45 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.04.02		subRC_SUCCESS subRC_SYS_UTIL_CONF_DATA_ALLOC_ERR

Table 45 Get Config Data Return Codes

Table 46 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
unsp__ConfigData (list)	• Name – The name of the RTP Parameters • Value – The value associated with the RTP Parameters.		Srx/Main/SvcAstsignedPerOfficeACR Srx/Main/CustomerType Srx/Main/SoftSwitchType Srx/Xdm/DnCode/digits/minLen Srx/Xdm/DnCode/digits/maxLen Srx/Xdm/DnCode/npa/minLen Srx/Xdm/DnCode/npa/maxLen Srx/Xdm/HomeDn/digits/maxLen Srx/Xdm/HomeDn/digits/minLen Srx/Xdm/OfficeCode/digits/minLen Srx/Xdm/OfficeCode/digits/maxLen Srx/Xdm/OfficeCode/countryCode/minLen Srx/Xdm/OfficeCode/countryCode/maxLen Srx/Sip/AuthEnabled Srx/Sip/max_number_of_sessions Srx/Sip/default_number_of_sessions Srx/Subp/Address Srx/Subp/Port Srx/Subp/NumberOfInstances Srx/Subp/OverLoadCheckPeriod Srx/Subp/JobCheckSleep Srx/Subp/NumObjectsToPauseAfter Srx/Subp/JobLoopSleepTime Srx/Subp/JobClearTime Srx/Subp/ClearJobDays Srx/Subp/ClearErrorDays Srx/Main/VMWIcapabilityDefault Srx/Main/InvalidStringChars Srx/Main/DisableCFDestValidation Srx/Main/AllowCFDestEmergency Srx/Main/SwitchFPRestrict Srx/Main/SrxSwitchLangMode Srx/Main/SrxDefaultLang Srx/Main/SrxSupportedLangList Srx/Main/SS7Variant Srx/Main/hiQVariant Srx/Main/ApplicationType Srx/Main/AuthTRSOverride Srx/Main/AuthTRSOverrideValues	

Table 46 Get Config Data Output Fields

7.4 Set Config Parm Data

This function is used to set an RTP Parameter or list of RTP Parameters.

Note that the list of RTP parameters that are provisionable via the SOAP interface is not the complete list of existing RTP parameters in the system, but rather is a limited set defined in the file RtpParameters.conf, which is delivered on the system in the /unisphere/srx3000/UNSPsubp directory.

If an RTP parameter is not included in this file then access to it via the SOAP interface is denied, even if the RTP parameter is defined as Customer type and is accessible via OpenScape Voice Assistant/CLI. This approach allows different configuration files to be provided for different solutions.

The format of each line of the file is as follows:

<parameterName> type: <type of parameter> endtype value: <accepted values of the parameter according to its type> endtype

The <type of parameter> can be any of the following: IpAddress, boolean, string, or integer, and string parameters contain the length in braces {}.

For string parameters, if a range of numeric values is allowed, the valid range is included in square brackets []; for example: [0-9]. If a range of alphabetical values is allowed, it is included in angled brackets <>; for example, < A-E >. If specific characters are allowed, these are defined in parentheses () separated by commas; for example: (*,#).

For integer parameters, if a range of values is allowed, the valid range is included in square brackets []; for example [0-180]. If specific values are allowed, these are defined in parentheses () separated by commas; for example: (1,2,3,6).

For Boolean parameters, the value should be defined as (true,false).

The following is an excerpt from the configuration file, showing several example lines:

```
Srx/Main/OwnCountryCode type: string { 0-3 } endtype
value: [ 0-9 ] endvalue
Srx/Main/RACFCfmTreat type: integer endtype value: [ 1-
2 ] endvalue
Srx/Main/SS7Variant type: integer endtype value: (
1,2,3,6 ) endvalue
Srx/Main/SvcAssignedPerOfficeACR type: boolean endtype
value: ( true,false ) endvalue
```

If a parameter in the configuration file does not have a validation rule defined, the operator cannot use the Set request to update the RTP parameter, but it is possible to retrieve the value of it using the Get request below.

For the Set operation, the SOAP Server validates that the requested parameter exists in the file and checks whether the value that has been passed is valid according to the rules defined in the file for this parameter.

Since a list of RTP parameters may be provided in the Set request, there may be one or more parameters that could not be processed for some reason; for example if the provided value is out of range. In this case, those entries are skipped and processing continues in an attempt to update the remaining parameters in the list. The response to the Set request includes a list of any parameters that were not successfully updated, and an error message describing the reason for the failure.

Table 47 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 47 Set Config Parm Data Return Codes

Table 48 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID inSection 5.4, “Primitive Data Definitions”, on page 66		
ConfigParmDataList	An array of ConfigData elements, each consisting of the following attributes:			
Name	The name of the RTP parameter.		Character String	required
Value	The value to which the RTP parameter is to be set.		Character String	required

Table 48 Get Config Parm Data Input Fields

7.5 Get Config Parm Data

This function is used to retrieve and display an RTP parameter or list of RTP parameters. Its functionality is similar to the GetConfigData interface, but with additional flexibility and additional available parameters to display.

Table 49 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 49 Get Config Parm Data Return Codes

Table 50 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID inSection 5.4, “Primitive Data Definitions”, on page 66		
ConfigDataNameList	An array of strings, each one being the name of the RTP parameter.		Character String	required

Table 50 Get Config Parm Data Input Fields

7.6 Get COS List

This method retrieves a list of the Class of Services that are defined on the system.

Table 51 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
1.0.5		subRC_INTERNAL_ERROR subRC_DB_UTIL_COS_LOOKUP_FAILED

Table 51 Get COS List Return Codes

Table 52 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			N/A

Table 52 Get COS List Input Field Descriptions

Table 53 lists the output fields for this method.

Field Name	Description			
COSList(List)	A string list with the names of the Class of Service.			
PaginatedInfo	Standard paging output structure. For details, refer to Section 5.5, “Paging”, on page 72.			

Table 53 Get COS List Output Fields

7.7 Get Rate Area List

This method returns the list of RateAreas that have been defined on the system.

Table 54 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
1.0.6		subRC_SUCCESS subRC_DB_UTIL_RATEAREA_LOOKUP_FAILED

Table 54 Get Rate Area List Return Codes

Table 55 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			N/A

Table 55 Get Rate Area List Input Field Descriptions

Table 56 lists the output fields for this method.

Field Name	Description			
GetRateArea ListData (List)	A string list with all the rate area lists defined.			
PaginatedInfo	Standard paging output structure. For details, refer to Section 5.5, “Paging”, on page 72.			

Table 56 Get Rate Area List Output Fields

7.8 Get Intercept List

This SOAP request provides the capability to display a list of currently existing Intercept IDs, as is also possible via the OpenScape Voice Assistant. For each entry, the Intercept Name is returned.

Table 57 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
05.03.01		subRC_SUCCESS subRC_DB_UTIL_INTERCEPT_LOOKUP_FAILED subRC_SYS_UTIL_INTERCEPT_ALLOC_ERR

Table 57 Get Intercept List Return Codes

Table 58 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
GetInterceptListData (list)	The names of the intercepts.	1..15 (each list element)	List of character strings	N/A

Table 58 Get Intercept List Output Fields

7.9 Get Calling Location List

This SOAP request provides the capability to display a list of existing Calling Locations. For each entry, the Calling Location is returned.

Table 59 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.01.00.34		subRC_SUCCESS subRC_DB_UTIL_CALLINGLOC_LOOKUP_FAILED

Table 59 Get Calling Location List Return Codes

Table 60 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, "Paging", on page 72.			N/A

Table 60 Get Calling Location List Input Field Descriptions

Table 61 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
GetCallingLocationListData (list)	The IDs of the calling locations.			N/A
PaginatedInfo	Standard paging output structure. For details, refer to Section 5.5, "Paging", on page 72.			

Table 61 Get Calling Location List Output Fields

7.10 Get Associated SIP Endpoint List

This SOAP request provides the capability to display a list of Associated SIP Endpoints. For each entry, the Associated SIP Endpoint Name is returned.

Table 62 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.01.01.16		subRC_SUCCESS subRC_INVLD_FEATURE_DATA subRC_SYS_UTIL_ASSOC_ENDPOINT_ALLOC_ERR subRC_INTERNAL_ERROR

Table 62 Get Associated Sip Endpoint List Return Codes

Table 64 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default Value
Endpoint Name	The name of the Endpoint.	16 char string		N/A

Table 63 Get Associated Sip Endpoint List Input Fields

Table 64 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
Endpoint Name	The name of the Endpoint.	string		
GetAssociatedSip EndpointListData(list)	The names of the SIP Endpoints.			N/A

Table 64 Get Associated Sip Endpoint List Output Fields

7.11 Get Licensing Info

This SOAP request provides the capability to display the following information:

- Licensing Information on the OpenScape Voice system for “End User” licenses.
- Licensing Information on the OpenScape Voice system for “Dynamic Session” licenses. It controls the number of SIP subscribers that are concurrently registered in the system.
- Licensing Information on the OpenScape Voice system for “Trucking” licenses. The Trunking Session licenses control the number of concurrent calls between two SIPQ endpoints.
- Licensing Information on the OpenScape Voice system for “Client Access” licenses.
- Licensing Information on the OpenScape Voice system for OSMO licenses. It controls the number of mobile devices subscribed to the system.
- For End User licenses, the data returned will be the “Installed” number of licenses and the count of the licenses that are currently being “Used”. For Dynamic Session, Trunking, Client access and OSMO licenses a variety of attributes are returned. Please refer to the returned data in the functions below for further details.

For End User licenses, the data returned will be the “Installed” number of licenses and the count of the licenses that are currently being “Used”. For Dynamic Session and Trunking licenses, a variety of attributes are returned. Please refer to the returned data in the functions below for further details.

The Licensing Manager component is used to retrieve this information.

Table 65 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
12.00.01.30		subRC_SUCCESS subRC_APP_LICENSE_MANAGER_ERROR

Table 65 Get Licensing Info Return Codes

Table 66 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Table 10 on page 66.			
InterfaceWSDL	The WSDL version being used by the requesting SOAP client.	For more information, refer to Section 5.1.1, "The Interface WSDL and Interface Release Control", on page 61.		
LicensingType	The type of licensing info being requested. Option AllLicense is used to retrieve the data of all license types in a single response.	enum	AllLicense EndUserLicense TrunkingSessionLicense DynamicLicense ClientAccessLicense OSMOLicense SLESLicense	EndUserLicense

Table 66 Get Licensing Info Input Fields

Table 67 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
LicensingType	The type of licensing info being returned.	enum	AllLicense EndUserLicense TrunkingSessionLicense DynamicLicense ClientAccessLicense OSMOLicense SLESLicense	N/A
InstalledCount	The number of licenses of the specified type currently installed on the switch. Note this attribute applies only to EndUserLicense type.	integer	any non-negative integer	N/A
UsedCount	The number of licenses of the specified type currently in use on the switch. Note this attribute applies only to EndUserLicense type.	integer	any non-negative integer	N/A
TrunkingLicensing Info	A structure containing the following attributes:			

Table 67 Get Licensing Info Output Fields (Sheet 1 of 4)

Field Name	Description	Length	Range	Default Value
(TrunkingLicensingInfo) AssignedTrunkingSessionLicenses	Number of trunking session licenses currently assigned.	integer	any non-negative integer	N/A
(TrunkingLicensingInfo) Average1hUsage	The average number of trunking session licenses used within the last hour.	integer	any non-negative integer	N/A
(TrunkingLicensingInfo) Maximum1hAverageUsage	The maximum value of the Average1hUsage attribute during the previous 24 hours.	integer	any non-negative integer	N/A
(TrunkingLicensingInfo) CustomerViolationLimit	The maximum number of times the trunk usage may exceed the number of trunking session licenses.	integer	any non-negative integer	N/A
(TrunkingLicensingInfo) CustomerViolationCounter	The number of times the trunk usage has exceeded the number of trunking session licenses.	integer	any non-negative integer	N/A
(TrunkingLicensingInfo)RemainingDays	Days until the license expires	integer	any non-negative integer	N/A
(TrunkingLicensingInfo) TrunkingLicenseManagerStatus	The status of the license manager.	enum	LMStatusNormal LMStatusInThreshold LMStatusOverloaded	N/A
DynamicLicenseInfo	A structure containing the following attributes:			
(DynamicLicenseInfo) Available	The number of available dynamic licenses.	integer	any non-negative integer	N/A
(DynamicLicenseInfo) InUse	The number of dynamic licenses currently in use.	integer	any non-negative integer	N/A
InUseMobile	The number of dynamic licenses currently in use for mobile subscriptions, when OSMO licensing is not installed in the system. When OSMO licensing is installed this counter is set to 0.	integer	any non-negative integer	N/A
(DynamicLicenseInfo) PreviousDayMaxUsed	The maximum value of the InUse attribute that has been reported during the last 24 hours.	integer	any non-negative integer	N/A

Table 67 Get Licensing Info Output Fields (Sheet 2 of 4)

Field Name	Description	Length	Range	Default Value
(DynamicLicenseInfo) CustomerViolationLimit	The maximum number of times the dynamic license usage may exceed the number of dynamic licenses.	integer	any non-negative integer	N/A
(DynamicLicenseInfo) CustomerViolationCounter	The number of times the dynamic license usage has exceeded the number of dynamic licenses.	integer	any non-negative integer	N/A
(DynamicLicenseInfo)RemainingDays	Days remaining before license expiration	integer	any non-negative integer	N/A
ClientAccessLicensingInfo	A structure containing the following attributes:			
(ClientAccessLicensingInfo) ClientAccessLicenses	The number of available client access licenses.	integer	any non-negative integer	N/A
(ClientAccessLicensingInfo) CurrentUsage	The number of client access licenses currently in use.	integer	any non-negative integer	N/A
(ClientAccessLicensingInfo) CustomerViolationLimit	The maximum number of times the client access license usage may exceed the number of client access licenses.	integer	any non-negative integer	N/A
(ClientAccessLicensingInfo CustomerViolationCounter	The number of times the client access license usage has exceeded the number of client access licenses.	integer	any non-negative integer	N/A
(ClientAccessLicensingInfoRemainingDays	Days remaining before license expiration	integer	any non-negative integer	N/A
OSMOLicenseInfo	A structure containing the following attributes:			
(OSMOLicenseInfo) Available	The number of available OSMO licenses. When OSMO licensing is not installed this counter is set to 0.	integer	any non-negative integer	N/A
(OSMOLicenseInfo) InUse	The number of OSMO licenses currently in use. When OSMO licensing is not installed this counter is set to 0.	integer	any non-negative integer	N/A
(OSMOLicenseInfo) Maximum24hUsage	The maximum value of the InUse attribute that has been reported during the last 24 hours. When OSMO licensing is not installed this counter is set to 0.	integer	any non-negative integer	N/A

Table 67 Get Licensing Info Output Fields (Sheet 3 of 4)

Field Name	Description	Length	Range	Default Value
(OSMOLicenseInfo) CustomerViolations	The maximum number of times the OSMO license usage may exceed the number of available OSMO licenses. When OSMO licensing is not installed this counter is set to 0.	integer	any non-negative integer	N/A
(OSMOLicenseInfo) CustomerViolationLimit	The maximum number of times the OSMO license usage has exceeded the number of available OSMO licenses. When OSMO licensing is not installed this counter is set to 0.	integer	any non-negative integer	N/A
(OSMOLicenseInfo) Enabled	A Boolean flag indicating whether OSMO licensing is installed in the system.	boolean	False, true	N/A
(OSMOLicenseInfo)RemainingDays	Days remaining before license expiration	integer	any non-negative integer	N/A
(SLESLicenseInfo	A structure containing the following attributes:			
(SLESLicenseInfo) RemainingDays	Days remaining before license expiration	integer	any non-negative integer	N/A

Table 67 Get Licensing Info Output Fields (Sheet 4 of 4)

7.12 Get Licensing Usage Information

SOAP server supports the retrieval of Voice, OSMO and 3rdParty Licensing Usage data.

Table 68 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
V6.00.28.01	N/A	subRC_SUCCESS

Table 68 Get Licensing Usage Codes

Table 69 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Table 10 on page 80.			
LicenseUsageData	The type of license usage data being requested. Option VoiceAllData is used to retrieve all usage data in a single response.	enum	VoiceAllData VoiceData VoiceUCData VoiceOSMOData	VoiceData

Table 69 Get Licensing Usage Input Fields

Table 70 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
LicenseUsageVoiceData	A structure containing the following attributes:			
(LicenseUsageVoiceData) BillingPeriodID	Billing periods are defined as monthly time intervals. The billing period ID is the number of the month where the billing period starts.	integer	any integer	N/A
(LicenseUsageVoiceData) HighWatermark	The maximum value set for used licenses. Any increment which produces a current value numerically greater than the current 'high watermark' should cause the 'high watermark' to increase to the current value.	integer	any integer	N/A
(LicenseUsageVoiceData) ProductInstanceID	An ID identifying the product instance.	Char array	0..40	N/A
(LicenseUsageVoiceData) RemainingDays	The number of remaining days until the license is expired.	integer	any integer	N/A
LicenseUsageOSMOData	A structure containing the following attributes:			
(LicenseUsageOSMOData) BillingPeriodID	Billing periods are defined as monthly time intervals. The billing period ID is the number of the month where the billing period starts.	integer	any integer	N/A
(LicenseUsageOSMOData) HighWatermark	The maximum value set for used licenses. Any increment which produces a current value numerically greater than the current 'high watermark' should cause the 'high watermark' to increase to the current value.	integer	any integer	N/A
(LicenseUsageOSMOData) ProductInstanceID	An ID identifying the product instance.	Char array	0..40	N/A
(LicenseUsageOSMOData) RemainingDays	The number of remaining days until the license is expired.	integer	any integer	N/A

Table 70 Get Licensing Usage Output Fields

The interface **unsp__GetLogFiles** has been implemented for attaching the log file tar created by the unsp__CollectLogFiles() and attach it to the SOAP envelope using SOAP with Attachments (SwA). SwA is not a W3c recommendation and it is superseded by Message Transfer Optimized

Mechanism (MTOM) which was suggested in the initial version of the LM. However current version of Axis does not support this technology yet, so it was decided to choose the more conventional DIME.

Along with the attachment the following parameters will be included in the SOAP response of **unsp__GetLogFiles**:

- **TotalNumberOfSuppressed** : Each time a new record that satisfies filtering criteria is found, then this number is increased by num_prior_occurences.
- **TotalNumberOfRecords** : this number is increased by one each time a record that satisfies input filtering criteria is found.
- **Distinct Records**: Each time a new record that satisfies filtering criteria is found, and its error description is not found in the pool of error descriptions then this generic counter is increased by 1.

The following table represents the input parameters:

Field Name	Description	Length	Range	Default Value
Operator Id				
Log File Name	Log File Name			

7.13 Get Destination List

This SOAP request provides the capability to display a list of existing destinations. For each entry, the name of the destination is returned.

Table 71 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.02.07		subRC_SUCCESS subRC_APP_UTIL_DEST_NP_NAME_LEN_INVALID subRC_SYS_UTIL_DEST_ALLOC_ERR subRC_DB_UTIL_DEST_LOOKUP_FAILED

Table 71 Get Destination List Return Codes

Table 72 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			N/A

Table 72 Get Destination List Input Field Descriptions

Table 73 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
DestinationListData(list)	The names of the destinations.			N/A
PaginatedInfo	Standard paging output structure. For details, refer to Section 5.5, “Paging”, on page 72.			

Table 73 Get Destination List Output Fields

7.14 Translate Dialed Digits

This method provides the ability to perform a test translation of a given dialed digit string from an existing subscriber on the system. It will test the routing definitions and facilities of the platform. This method will return the platform address translation and routing destination for a subscriber dialing a specific set of digits. The BG Name is passed as input, but not currently used. It may be used in the future to allow for non-DID subscribers.

The result structure contains two sets of return information. The first is that the request was proper and could be processed. The second (the Translation Result) is the results of the translation itself.

Table 74 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.04.04		subRC_SUCCESS subRC_APP_SUB_DN_DOES_NOT_EXIST subRC_DB_SUB_DN_LOOKUP_FAILED

Table 74 Translate Dialed Digits Return Codes

Table 75 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default Value
ServiceId	Please refer to Table 10 on page 66.			
DialedDigits	The DN of the potential terminating destination.	30	Numeric string	“”

Table 75 Translate Dialed Digits Input Fields

Table 76 lists the *output* fields for this method.

Field Name	Description	Length	Range/Value	Default Value
Translation Result	The result code returned by XLA for the attempted translation.	enum	XLARC_Success, XLARC_Failure, XLARC_Error, XLARC_InvalidAddr, XLARC_InvalidAddrFormat, XLARC_NotEnoughDigits, XLARC_InvalidCAC, XLARC_InvalidCACFormat, XLARC_UnauthorizedCarrTraffic, XLARC_FGDInScreeningFailed, XLARC_AllRoutesBusy, XLARC_NoRoutesFound, XLARC_InvalidIntcptId, XLARC_UnallocatedNumber, XLARC_DNDisconn, XLARC_Enum, XLARC_Misdialing, XLARC_NoIntcptTrt, XLARC_EPUnregistered, XLARC_EmptyNpId, XLARC_ErrorConvertType	N/A
DestType	The destination type as determined by the translation, if successful.	enum	XLADT_None, XLADT_Dest, XLADT_EP, XLADT_Svc, XLADT_Ln, XLADT_TrkCircuit, XLADT_MGCPMS, XLADT_Vacant, XLADT_HomeDN, XLADT_DNDisconn, XLADT_Enum, XLADT_Intcpt, XLADT_InvalidCode, XLADT_TOD, XLADT_E164, XLADT_NpId, XLADT_HomeExt, XLADT_HomeLoc, XLADT_EmrgLoc, XXLADT_CallingLoc, XLADT_Prefix_Dest	N/A
DestName	The destination name as determined by the translation, if successful.			N/A

Table 76 Translate Dialed Digits Output Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Value	Default Value
CallType	The call type as determined by the translation, if successful.	enum	XLACT_None, XLACT_CallNonCarrier, XLACT_CallLECIntraLATA, XLACT_CallLECOptr, XLACT_CallLECOptrIntraLATA, XLACT_CallCarrIntraLATA, XLACT_CallCarrInterLATA, XLACT_CallCarrIntl, XLACT_CallCarrIntlZone1, XLACT_CallCarrOptr00, XLACT_CallCarrCutThruFGPD, XLACT_CallCarrFGPB, XLACT_CallSvc, XLACT_CallCarrOptrIntraLATA, XLACT_CallCarrOptrInterLATA, XLACT_CallCarrOptrIntl, XLACT_CallCarrOptrIntlZone1, XLACT_CallCarrTollFree, XLACT_CallDA411, XLACT_CallDAHNPA, XLACT_CallDAFNPA, XLACT_CallEmrg, XLACT_CallLECOptrLocal, XLACT_CallCarrLD	N/A
DialingPattern	The dialing pattern as determined by the translation, if successful.	enum	XLADP_None, XLADP_Optr0Plus, XLADP_Optr0, XLADP_Optr00, XLADP_IntlOptr, XLADP_IntlDDD, XLADP_DDD, XLADP_NP, XLADP_FGPB, XLADP_FGPD, XLADP_TollFreeCode, XLADP_CACHome, XLADP_CACHomeNoCarrDesig, XLADP_CACHomeCarrDesig, XLADP_OffNetAC, XLADP_OnNetAC, XLADP_FacilityAC, XLADP_AttdntAC, XLADP_Extension	N/A

Table 76 Translate Dialed Digits Output Fields (Sheet 2 of 2)

7.15 Get Supported Language List

This function is used to retrieve and display language information, including the list of languages supported on the system.

Table 77 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01.06	N/A	subRC_SUCCESS subRC_APP_UTIL_LANG_SWITCH_MODE_INVALID subRC_APP_UTIL_LANG_SWITCH_LOOKUP_FAILED subRC_APP_UTIL_LANG_DEFAULT_INVALID subRC_APP_UTIL_LANG_INVALID

Table 77 Get Supported Language List Return Codes

There are no input fields for this method.

Table 78 lists the output fields for this method.

Field Name	Description	Length	Range/Values	Default Value
SwitchLanguageMode	An array of strings, each one being the name of the RTP parameter.	enum	Single Multiple	N/A
SystemDefaultLanguage	The default language for the switch.	1..30	Character String	N/A
GetSupportedLanguageListData	An array of GetSupportedLanguageListEntry elements, each containing the following fields:			
GetSupportedLanguageListEntry. Language	The name of a supported language.	1..30	Character String	N/A
GetSupportedLanguageListEntry. LanguageCode	The three digit code used to represent the supported language.	1..3	Character String	N/A

Table 78 Get Supported Language List Output Fields

7.16 Get Total Number of Call Pickup Groups

This function is used to calculate the number of call pick up groups among all BGs.

Call pick up groups with the same ID that belong to different BGs are counted as independent entities. Call pick up groups of the same Id that are assigned to multiple subscribers that belong to the same BG are considered as a single entity. In V3.1, the function is added under utilities menu (80) option 8o.

Table 79 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
12.00.01.32	N/A	subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED,

Table 79 Get Total Number of Call Pickup Groups Return Codes

Table 72 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorID	The operator issuing the command	Please refer to the definition of Operator ID in Section 5.4, “Primitive Data Definitions”, on page 66.		
BGName	The name of BG group	When NIL, ALL BG groups are included	Character String	Optional

Table 80 Get Total Number of Call Pickup Groups Input Fields

Table 73 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
CPUCount	The operator issuing the command		Integer	N/A
BGName	The name of BG group that contains the number of CPU groups. When NIL, ALL BG groups are included for the calculation		Character String	

Table 81 Get Destination List Output Fields

7.17 Get Fully Qualified Number

This interface is used in order to convert a specified sequence of digits (DialedDigits) to a FQN (Fully Qualified Number) or CPN (Complete Private Number) in the context of the InputFQN Numbering Plan.

Table 85 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
12.00.04.08	N/A	subRC_SUCCESS Success subRC_APP_SUB_FQN_INFQN_INVALID Invalid Fully Qualified Number (input) subRC_APP_SUB_FQN_DIALEDDIGITS_INVALID Invalid Dialed Digits (input) subRC_APP_SUB_FQN_RSS_INFQN_DOES_NOT_EXIST Input fully qualified number not found in RSS subRC_APP_SUB_FQN_RSS_INFQN_INVALID Input fully qualified number invalid subRC_APP_SUB_FQN_XLA_TRANSLATION_INCOMPLETE Translation of dialed digits from XLA not completed subRC_APP_SUB_FQN_XLA_TRANSLATION_FAILED Translation of dialed digits from XLA failed subRC_APP_SUB_FQN_RSS_TRANSLATED_DN_DOES_NOT_EXIST Translated number not found in RSS subRC_APP_SUB_FQN_RSS_TRANSLATED_DN_INVALID Translated number invalid subRC_APP_SUB_FQN_NDAL_CONVERTION_FAILED Conversion from NDAL to fully qualified number failed

Table 82 Get Fully Qualified Number Return Codes

Table 86 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
InputFQN	The directory number that has the numbering plan defined in the OpenScape Voice system.	32	character string	required
DialedDigits	The directory number that would be converted to FQN or CPN.	32	character string	required

Table 83 Get Fully Qualified Number Input Fields

Table 87 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
Result	The result of the operation.		Type unsp__ResultCodeStruct	
FullyQualifiedNumber	The number in FQN or CPN format	32	Character String	

Table 84 Get Fully Qualified Number Output Fields

7.18 Get System Info

This interface is used in order to retrieve basic information on the system such as CPU, memory usage and if the system is virtualized or not.

Table 85 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
		subRC_SUCCESS Success Others ??

Table 85 Get System Info Return Codes

Table 86 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required

Table 86 Get System Info Input Fields

Table 87 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
Result	The result of the operation.		Type unsp__ResultCodeStruct	
OvlMgrSysInfo	A structure that consists of the following:		String	
CpuUsed	Amount of CPU usage		String	
MemoryUsed	Amount of memory usage		String	
UpdateTime	Time of last update		Sring	
BootTime	When system was last booted		String	
Virtualized	is the system virtual		Boolean	

Table 87 Get System Info Output Fields

8 Network Management Interfaces via SOAP

For the requests in this section, note that query operations typically include a parameter called “Specific”. This is a Boolean parameter that is used to indicate whether a specific (single) object is being requested, or a list of objects is being retrieved. If True, any other input parameters are used to search for the exact object instance being requested. If False, any other input parameters are used as filter criteria and a list of objects matching the given parameters may be returned. The default if the parameter is not passed is False.

8.1 Office Code

8.1.1 Create Office Code

This method creates an office code. Table 88 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01	N/A	subRC_SUCCESS

Table 88 Create Office Codes Return Codes

Table 89 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OfficeCodeId	Numeric identifier of a specific office code. Must refer to an existing office code.	int		0
OfficeCode	The symbolic name of the office code reference.		Must be a digit sequence of maximum length 9. N. America: 6 digits Japan: 5 digits Korea: 4 or 5 digits Europe: 1 to 9 digits	
CountryCode Length	Length of the country code prefix in the office code.	int	0 – 4 in all markets	0
AreaCode Length	Length of the area code prefix in the office code.	int	N.America: 3 Japan: 1-5 Korea: 1-2 China: 2-3 Europe: 0-5	
Attributes	Defines attributes of the office code. Enumerated type.		AtrNo, AtrYes	

Table 89 Create Office Codes Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
ChargeArea	The charge area string (Japan market only). In the Japan market, **ChargeArea** must be provided; otherwise, it must be an empty string.	5-digit string		EmptyStr.

Table 89 Create Office Codes Input Fields (Sheet 2 of 2)

8.1.2 Modify Office Code

This method modifies an office code. Table 90 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01	N/A	subRC_SUCCESS

Table 90 Modify Office Code Return Codes

Table 91 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OfficeCodeId	Numeric identifier of a specific office code. OfficeCodeId must refer to an existing office code.	int		0
OfficeCode	The symbolic name of the of the office code reference.		Must be a digit sequence of maximum length 9. N. America: 6 digits Japan: 5 digits Korea: 4 or 5 digits Europe: 1 to 9 digits	
CountryCode Length	Length of the country code prefix in the office code.	int	0 – 4 in all markets	0
AreaCode Length	Length of the area code prefix in the office code.	int	N.America: 3 Japan: 1 – 5 Korea: 1 – 2 China: 2 – 3 Europe: 0 – 5	
Attributes	Defines attributes of the office code. Enumerated type.		AtrNo, AtrYes	
ChargeArea	The charge area string (Japan market only). In the Japan market, ChargeArea must be provided; otherwise, it must be an empty string.	5-digit string		EmptyStr

Table 91 Modify Office Code Input Fields

8.1.3 Delete Office Code

This method deletes an office code. Table 92 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01	N/A	subRC_SUCCESS

Table 92 Delete Office Code Return Codes

Table 93 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OfficeCodeId	Numeric identifier of a specific office code. OfficeCodeId must refer to an existing office code.	int		0
OfficeCode	The symbolic name of the of the office code reference.		Must be a digit sequence of maximum length 9. N. America: 6 digits Japan: 5 digits Korea: 4 or 5 digits Europe: 1 to 9 digits	

Table 93 Delete Office Code Input Fields

8.1.4 Display Office Code

This method retrieves an office code and displays all information related to it. Table 94 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01	N/A	subRC_SUCCESS

Table 94 Display Office Code Return Codes

Table 95 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
Specific	Refer to the paragraph at the beginning of Chapter 8.			
Paging	Standard paging structure for paged queries. For details refer to Section 5.5, “Paging”, on page 72.			Null
OfficeCodeId	Numeric identifier of a specific office code. OfficeCodeId must refer to an existing office code.	int		0

Table 95 Display Office Code Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
OfficeCode	The symbolic name of the of the office code reference.		Must be a digit sequence of maximum length 9. N. America: 6 digits Japan: 5 digits Korea: 4 or 5 digits Europe: 1 to 9 digits	
Prefix	Range of office codes.			EmptyStr
Destination Type	Type of destination; used only in query. If given, restricts the search to office codes with that type.		DtUnused**:** includes all office codes. Other destination types are not used.	Optional
Filter	Condition that must be held for retrieved office codes. It is used for filtering office codes retrieved in query.		**All**: include all office codes. **DnDnNoCodeReference:** objects with no references in e164_code_t; **DnDnCodeReference:** objects with at least one reference in **e164_code_t;** **DnHomeDn**: objects with no references in **e164_dn_t** **DnHomeDn**: objects with at least one reference in e164_dn_t **DnNotImported**: Objects that do not set the IMPORT field. **DnImported**: objects that do set the IMPORT set.	All
Filter, *cont.*			**DnVacant**: objects that have at least one home DN whose destination type is VACANT **DnNotVacant**: objects that have no home DN whose destination type is VACANT	All
CountActivate	Indicates whether counts are retrieved.		CmdUnassigned CmdYes CmdNo	CountActivate
OfficeCode Sort Type	Defines whether Sorting is performed based on OfficeCode ot not	Enum	OfficeCodeStUnassigned, OfficeCodeStOfficeCode	OfficeCodeStUnassigned
SortOrder	Determines the way the objectsare sorted.	Enum	SortOrderAscending - Objects aresorted in ascending order. SortOrderDescending - Objects are sorted in descending order.	SortOrderAscending

Table 95 Display Office Code Input Fields (Sheet 2 of 2)

8.2 Home DN

8.2.1 Create Home Dn

This method creates a HomeDN. Table 96 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01	N/A	subRC_SUCCESS

Table 96 Create Home DN Return Codes

Table 97 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
StartCode	Mandatory digit string. Indicates the first subscriber number in range.	6	N. America: 4 digits Japan: 4 digits Korea: 4 digits Europe: 3 – 6 digits	EmptyStr
EndCode	Optional digit string that indicates the last subscriber number in range. If provided, it must meet the same conditions as startCode. The digit string must be greater than or equal to that of startCode.		N. America: 4 digits Japan: 4 digits Korea: 4 digits Europe: 3 – 6 digits	EmptyStr
OfficeCodeId	Numeric identifier of a specific office code. It must refer to an existing office code.	int		0
OfficeCode	Symbolic name of the of the office code reference.		Must be a digit sequence of maximum length 9. N.America: 6 digits Japan: 5 digits Korea: 4 or 5 digits Europe: 1 – 9 digits	

Table 97 Create Home DN Input Fields

8.2.2 Modify Home Dn

This method modifies a HomeDN. Table 98 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01	N/A	subRC_SUCCESS

Table 98 Modify Home DN Return Codes

Table 99 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
ModifyType	Used to identify the modify command.		MtModify MtAssign MtUnassign	
StartCodeId	Numeric identifier of a specific StartCode. The first Home DN to be modified is identified by the StartCodeId (or by the StartCode).	int		
StartCode	Mandatory digit string. The first Home DN to be modified is identified by the StartCode (or by the StartCodeId).	6	N. America: 4 digits Japan: 4 digits Korea: 4 digits Europe: 3 to 6 digits	EmptyStrs
EndCodeId	Numeric identifier of a specific EndCode. The last Home DN to modify is identified by the EndCodeId (or by the EndCode).	int		
EndCode	Optional digit string. The last Home DN to modify is identified by the EndCode (or by the EndCodeId). If provided, it must refer to an existing Home DN that represents a continuous sequence.		N. America: 4 digits Japan: 4 digits Korea: 4 digits Europe: 3 to 6 digits	
OfficeCodeId	Numeric identifier of a specific office code. Must refer to an existing office code.	int		
OfficeCode	The symbolic name of the office code reference. The office code is identified either by OfficeCode or OfficeCodeId.		Must be a digit sequence of maximum length 9. N. America: 6 digits Japan: 5 digits Korea: 4 or 5 digits Europe: 1 to 9 digits	
DestinationId	Numeric identifier of a destination.	int		0
Destination Name	Name of the destination associated with this Home DN.	str		EmptyStr
Destination Type	Enumerate value that represents the type of destination. Mandatory. The values that it can take vary for different markets.		DtDest DtSvc DtHomeDn DtEnum DtTod DtIntcpt DtInvalidCode DtHomeLoc DtHomeExt DtPri DtUnused DtTrkCircuit DtVacant	

Table 99 Modify Home DN Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
SubscriberId	Numeric identifier of the subscriber profile.	int		
Subscriber Name	Symbolic name of the subscriber profile.	str		
NewDn	NewDn is a digit string of maximum length 15. It provides the new DN code with DNCHANGED destination. Used only when the ModifyType is set to UNASSIGN.	str		EmptyStr
EnableAssign	Applies only to the North America market. Optional. If marked as **No** and one or more of the home DNs has ExportTrgLnp, then the command is rejected.		CmdUnassigned CmdNo CmdYes	No
EnableMatch	Indicates whether the assignment of a line requires the digits of the line and home DN objects to match.		EmNo EmYes	True

Table 99 Modify Home DN Input Fields (Sheet 2 of 2)

8.2.3 Delete Home Dn

This method deletes a HomeDN. Table 100 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01	N/A	subRC_SUCCESS

Table 100 Delete Home DN Return Codes

Table 101 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
StartCodeId	Numeric identifier of a specific StartCode. The first Home DN to be deleted is identified by either the StartCodeId or the StartCode.	int		0
StartCode	Digit string. The first Home DN to be deleted is identified by either the StartCode or the StartCodeId.	6	N. America: 4 digits Japan: 4 digits Korea: 4 digits Europe: 3 to 6 digits	EmptyStr
EndCodeId	Numeric identifier of a specific EndCode. The last Home DN to delete is identified by either the EndCodeId or the EndCode.	int		0

Table 101 Delete Home DN Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
EndCode	Optional digit string. The last Home DN to delete is identified by either the EndCode or the EndCodeId. If provided, it must refer to an existing Home DN that represents a continuous sequence.		N. America: 4 digits Japan: 4 digits Korea: 4 digits Europe: 3 to 6 digits	EmptyStr
OfficeCodeId	Numeric identifier of a specific office code. OfficeCodeId must refer to an existing office code.	int		0
OfficeCode	It is the symbolic name of the office code reference. The office code is identified either by OfficeCode or OfficeCodeId.	9	N. America: 6 digits Japan: 5 digits Korea: 4 or 5 digits Europe: 1 to 9 digits	EmptyStr

Table 101 Delete Home DN Input Fields (Sheet 2 of 2)

8.2.4 Query Home Dn

This method queries a HomeDN. Table 102 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01	N/A	subRC_SUCCESS

Table 102 Query Home DN Return Codes

Table 103 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
Specific	Refer to the paragraph at the beginning of Chapter 8.			
Paging	Standard paging structure for paged queries. For details, refer to Section 5.5, “Paging”, on page 72.			Null
SortType	Determines whether the objects are retrieved with a 2-index or 4-index sort.	• StReference – Objects are sorted by officeCode and code. • StDestnation – objects are sorted by destinationType, destinationId, officeCode, and code.		StReference
SortDirection	Determines the way the objects are sorted.	• SortOrderAscending - Objects are sorted in ascending order. • SortOrderDescending - Objects are sorted in descending order.		SortOrderAscending
OfficeCode Prefix	Substring (or full string) identifying one or a set of office codes. Supports wildcarding (%).	str		

Table 103 Query Home DN Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
StartCode	Digit string. The first HomeDN to be retrieved is identified by StartCode. This field supports wildcarding when a specific OfficeCode Prefix is given.	6	N. America: 4 digits Japan: 4 digits Korea: 4 digits Europe: 3 to 6 digits	Emptystr
EndCode	Optional digit string. The last Home DN to retrieve is identified by the EndCode. It must refer to an existing HomeDN and is used in combination with StartCode to retrieve a range of HomeDns. It does not support wildcarding sequence.		N. America: 4 digits Japan: 4 digits Korea: 4 digits Europe: 3 to 6 digits	Emptystr
Destination Type	If provided, restricts the search to home DNs with that type.		DtDest DtEp DtLn DtSvc DtHomeDn DtEnum DtTod DtIntcpt DtInvalidCode DtHomeLoc DtHomeExt DtPri DtUnused DtTrkCircuit DtVacant	
Destination Name	If provided, further restricts the search to home DNs with that specific destination.	str		EmptyStr
Filter	Condition that must hold for retrieved home DNs. Used for filtering office codes retrieved in Query.		**DnAll**: Retrieve all home DNs **DnVacant:** objects whose destination has type VACANT. **DnNotVacant:** objects whose destination type is not VACANT. **Note:** This is a subset of the complete list of filters. Only these values are valid for Query home DN.	DnAll
BGName	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66 for a detailed description. A string containing the wildcard character (%) can also be the input. Matching entries will be returned in this case			

Table 103 Query Home DN Input Fields (Sheet 2 of 2)

8.3 Endpoint

8.3.1 Create Endpoint

This method creates an endpoint. Table 104 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
V5.00.18.01		subRC_APP_DSA_LOCATION_DOMAIN_NAME_SIP_PROXY_INVALID
10.00.01	N/A	subRC_SUCCESS

Table 104 Create Endpoint Return Codes

Table 105 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
Name	CLI name of the endpoint.	str		
IpAddress	Character string with the IP Address of the endpoint. If the IP address and port are provided with non-default value, it must be unique in the set of addresses that include: {ipAddress, port} combinations for endpoint {rasIpAddress,rasPort} combinations for endpoint {ipAddress, port} combinations for gatekeepers	str		“”
AltIpAddress	Optional secondary IP address of the endpoint. The same uniqueness rule (as above in IpAddress) applies in AltIpAddress.	str		
RasIpAddress	IP address of the RAS channel for the SIP endpoint.	str		
ManagementIpAddress	Character string for a separate address for the endpoint allowing its management using a different IP address for its configuration than the signaling address (IpAddress field).The ManagementIpAddress can be provisioned in IPv4, IPv6 or FQDN format. If this field is not configured, the IpAddress of the	str		“”

Table 105 Create Endpoint Input Fields (Sheet 1 of 12)

Field Name	Description	Length	Range	Default Value
Register Process	RTP name of the process that sends an RRQ request.			
Port	Port is the signaling address of the endpoint. If the IP address and port are provided with non-default value, it must be unique in the set of addresses that include: {ipAddress, port} combinations for endpoint {rasIpAddress,rasPort} combinations for endpoint {ipAddress, port} combinations for gatekeepers	int		0
AltPort	Optional secondary IP address of the endpoint.	int		
RasPort	RasPort is an integer that represents the RAS port number at the endpoint.	int		
EndPointType	Enumerated type that represents the type of the endpoint.	int	EtUnassigned EtSip	
Registration	An integer that represents bitwise assignments of five attributes: Operational state of the endpoint (highest 2 bits): UNNASIGNED, ACTIVE, BLOCKED_NEW, BLOCKED_ALL Administrative state of the dynamic endpoint (next 2 bits): UNNASIGNED, ACTIVE, BLOCKED_NEW, BLOCKED_ALL Authentication pre-granted: indicates whether authentication is pregranted for a dynamic endpoint (next 3 bits): UNNASIGNED, NEITHER_SIDE, A_SIDE, B_SIDE, BOTH_SIDES RRQ state: indicates whether a dynamic endpoint is (URQ, RRQ) (next 2 bits): UNNASIGNED, UNREGISTERED, REGISTERED RRQ type: indicates whether the endpoint is static or dynamic (lowest 2 bits): UNNASIGNED, STATIC, DYNAMIC_PORT (static IP), DYNAMIC	int	Default values: **Operstate**: UNNASIGNED=0 **adminState:** ACTIVE=1 **authGranted:** NEITHER_SIDE=1 If the endpoint is static: **rrqState:** UN-REGISTERED=1 **rrqType:** STATIC=1 If the endpoint is dynamic: **rrqState:** REGISTERED=2 **rrqType:** DYNAMIC =3	
TimeToLive	Expiration time for the endpoint (if dynamic).	int		

Table 105 Create Endpoint Input Fields (Sheet 2 of 12)

Field Name	Description	Length	Range	Default Value
RrqId	Internal numeric identifier used by the H.323 process to identify the registered endpoint.	Int		
AccountAuth	An enumeration of values for endpoint account authorization. Notes: This field is not modifiable after creation. ccountAuthBasedOnSub applies only to subscriber type endpoints.	enum	AccountAuthUnused, AccountAuthUnassigned, AccountAuthBasedOnSub, AccountAuthBasedOnEp, AccountAuthNoServices	AccountAuthUnused
ProfileId	Numeric identifier of the associated endpoint profile object.	int		
ProfileName	Symbolic name of the association endpoint profile object.	str		
Sessions	Maximum number of sessions allowed for the endpoint. Note that for non-PBX subscriber endpoints, this represents the number of messaging sessions, not the number of voice (audio) sessions.	Int	For SIP endpoint, it has a default value 3. For non-SIP endpoint, it takes the default value 1. Otherwise, it must take a value greater than 0 and less than a maximum that is provided as an RTP configuration variable. Default and maximum values are determined by the RTP parameters: Srx/Sip/ default_number_of_sessions and Srx/Sip/max_number_of_sessions.	3
Transport	Transport specifies the transport protocol for a SIP endpoint	int	One of TtUdp, TtTcp, and TtTls	
SdesSrtpSupport	"Best-Effort SRTP with SDES" attribute to determine whether to include the SRTP media in the SDP offer or not.	Enum	SrtpUnassigned SrtpAutomatic SrtpEnabled SrtpDisabled	SrtpDisabled
AnatSupport	Indicates whether the endpoint supports ANAT. AnatOn indicates that the endpoint supports ANAT. AnatOff means that the endpoint does not support ANAT. AnatAuto means whether the endpoint supports ANAT shall be determined by the SIP registrar during registration of the endpoint. AnatUnassigned indicates that the default value should be assigned,	Enum	AnatUnassigned AnatAuto AnatOn, AnatOff	If the registration of the endpoint is static: AnatOn If the registration of the endpoint is dynamic: AnatAuto (For registration types please refer to Endpoint Registration field)
Attributes	A list of the following Boolean flags:			

Table 105 Create Endpoint Input Fields (Sheet 3 of 12)

Field Name	Description	Length	Range	Default Value
(Attributes) SendAuthNumAs PAI	Indicates whether the AuthNum is send as PAI	boolean	false true	false
(Attributes) CallingPartyCateg ory	Indicates whether for this endpoint, the incoming CallingPartyCategory (CPC) is interpreted as the originating class of service.	boolean	false true	false
(Attributes) UpdateOnConfirm edDialaouge	Indicates whether this endpoint uses UPDATE for confirm dialogue support. If false, only SIP:RE-INVITE is used for confirm dialogue support.	boolean	false true	false
(Attributes) SendProvRespon seOnUpdate	Indicates whether the endpoint sends a provisional response during session updates.	boolean	false true	false
(Attributes) QSigOverSip	Indicates whether this endpoint supports SIPQ signaling.	boolean	false true	false
(Attributes) ImsFor3GPP	Indicates whether the endpoint supports 3GPP IMS. If switch application type is set to IMS MGCP then SIPSM constructs RequestURI in TEL URI format, and TO, FROM and P-Asserted headers host part are constructed with IMS domain name. Note that this attribute is not associated with a SIP subscriber, but with the SIP endpoint, so it can only be set (including being exported and imported) using the Endpoint provisioning requests, not the Subscriber requests.	boolean	false true	false
(Attributes) MTPEnabled	Indicates that the endpoint has precondition signaling enabled.	boolean	false true	false
(Attributes) EarlySessionMedi a	Indicates whether the endpoint has early-session media signaling enabled.	boolean	false true	false
(Attributes) TrustedSubscriber	Indicates whether the subscriber is located behind a trusted endpoint.	boolean	false true	false
(Attributes) GSMSubscribe	Indicates whether the endpoint is a GSM Mobile IP Centrex subscriber.	boolean	false true	false
(Attributes) SurvivableProxy	Indicates whether this endpoint is a survivable SIP proxy endpoint.	boolean	false true	false

Table 105 Create Endpoint Input Fields (Sheet 4 of 12)

Field Name	Description	Length	Range	Default Value
(Attributes) URISupportRequired	Indicates whether TEL URI must be used for sending addresses on this interface.	boolean	false true	false
(Attributes) URISupportAllowed	Indicates whether TEL URI is allowed to be used for sending addresses on this interface.	boolean	false true	false
(Attributes) SendUriAsReceived	Indicates whether TEL URI address specification is sent as originally received.	boolean	false true	false
(Attributes) SendUriInTelephoneSubscriberFormat	Indicates whether the TEL URI is sent in telephone subscriber (global number or local number) format (including phone context for local numbers).	boolean	false true	false
(Attributes) SendUriWithoutPhoneContext	Indicates whether the TEL URI is send in local number format without phone context. Note that the default is true, and this is the only currently supported option for local number format.	boolean	false true	true
(Attributes) SendUriAsDialString	Indicates whether the TEL URI is send in local number format with phone context. Note that the default is false, and this option is not currently supported.	boolean	false true	false
(Attributes) NumericUserIP	Indicates whether or not to treat all numeric user info (without user=phone) as user=ip.	boolean	false true	false
(Attributes) TruncateMIME	Indicates whether the size of the MIME will truncated to reduce its size.	boolean	false true	false
(Attributes) NationalOrInterNationalPrefix	Indicates whether national or international prefix should be inserted.	boolean	false true	false
(Attributes) PDCSBillingInfoSupported	Indicates whether the sending of the STP P-DCS-Billing-Info header parameters is supported.	boolean	false true	false
(Attributes) GenerateReasonHeader	Indicates whether for this endpoint a reason header is generated in response for responses/release messages without a reason header.	boolean	false true	false

Table 105 Create Endpoint Input Fields (Sheet 5 of 12)

Field Name	Description	Length	Range	Default Value
(Attributes) SendReasonHeaderForQ850	Indicates whether for this endpoint a reason header is sent for Q.850 protocol - non-2XX final responses mapped to Q.850 protocol specifications are allowed to be sent.	boolean	false true	false
(Attributes) SendReasonHeaderForSip	Indicates whether for this endpoint a reason header is sent for SIP protocol - non-2XX final responses which are SIP-specific (without Q.850 protocol specifications) are allowed to be sent.	boolean	false true	false
(Attributes) DoNotSendReasonHeader	Indicates whether for this endpoint a reason header is not to be sent - provides backward compatibility by filtering the reason header from any response sent to a SIP UA.	boolean	false true	false
(Attributes) SipOliRequired	Indicates whether for this endpoint the OLI is to be sent. The format of the OLI to be sent depends on the IsupOliRequired attribute. When WSDL is V5.00.22.01 or less this value is created/ modified/displayed. When WSDL is greater than V5.00.22.01, it is created/ modified/displayed as VoiceMail	boolean	false true	false (Supported until V5.00.22.01)
(Attributes) IsupOliRequired	Indicates whether the OLI is to be signaled as an ISUP_OLI parameter or as an OLI parameter.	boolean	false true	false
(Attributes) IgnoreIngressCpcOli	Indicates whether for this endpoint, the CPC, OLI, or ISUP OLI is ignored if received in the incoming SIP:INVITE.	boolean	false true	false
(Attributes) IgnoreAnswerForAnnouncement	Indicates whether the Answer for announcement should be ignored	Boolen	false true	false
(Attributes) EndpointIsSecure	Indicates whether security is enabled for this endpoint.	boolean	false true	false

Table 105 Create Endpoint Input Fields (Sheet 6 of 12)

Field Name	Description	Length	Range	Default Value
(Attributes) EndpointIsProxy	Indicates whether this endpoint is a SIP proxy server. Note that this attribute applies only to non-subscriber endpoints.	boolean	false true	false
(Attributes) RouteViaProxy	Indicates whether all SIP:INVITE messages to SIP UAs that are registered via an associated SIP endpoint are sent to the associated SIP endpoint. Note that this attribute may be true only if EndpointIsProxy is also true.	boolean	false true	false
(Attributes) EndpointIsInaccessible	Indicates whether this endpoint is in an inaccessible state. Note that this attribute is read-only and is only listed here for completeness.	boolean	false true	false
(Attributes) MediaServerRedirectionAllowed	Indicates whether media redirection is allowed for this endpoint.	boolean	false true	false
(Attributes) RerouteToAlternateIp	Indicates whether, in the scenario where the first destination is not reachable, the switch will look for an alternate destination for the new request.	boolean	false true	false
(Attributes) UseServerVirtualIp	Indicates whether to use the server virtual IP address for SIP gateways, SIP application server, and SIP voice mail. If TLS is selected as transport type then server to server communication is through mutual authentication.	boolean	false true	false
(Attributes) StaticIMSSub	Indicates whether this is the endpoint of a static IMS subscriber.	boolean	false true	false
(Attributes)CallCenterApplication	Indicates that the endpoint is a call center application.	boolean	false true	false
(Attributes)EndpointIsAssociatedServer	A SIP NNI endpoint assigned with this attribute set to true shall be excluded from Trunking Session licensing scenarios.Note that this attribute applies only in the Enterprise environment.	boolean	false true	false

Table 105 Create Endpoint Input Fields (Sheet 7 of 12)

Field Name	Description	Length	Range	Default Value
(Attributes)SendInsecureRefferedBy Header	When a Referred-By header field is included in the initial request sent to the transfer target (for a blind transfer), the Referred-By header (including any generic-parameters) will be passed transparently from the transferor to the transfer target.	boolean	false true	true
(Attributes)SendPubNrInFQNInFrom Header		boolean	false true	
(Attributes) OverrideIRMCodecRestrictions	This attribute forces selection of codecs by overriding IRM restrictions.	Boolean	false true	false
(Attributes) ReroutingSiptoSip	This attribute indicates whether the subscriber rerouting of incoming calls through the SIP endpoint to a survivable SIP subscriber is allowed.	Boolean	false true	false
(Attributes) ReroutingFwd	This attribute indicates whether subscriber rerouting of incoming calls through the SIP endpoint that are forwarded to a survivable SIP subscriber is allowed.	Boolean	false true	false
(Attributes) TransferHandoff	If this attribute is set, calls can be transferred to the Genesys.	Boolean	false, true	false
(Attributes)AllowUnregisterStaleRegistrations		Boolean	false, true	
(Attributes) Reserved4	The purpose for this is to be prepared for future urgent requests requiring a new endpoint attribute and where new functionality via 8K (e)patch is delivered but there is no time to deliver the corresponding Assistant changes at the same time. The intention is to use these reserve attributes only in exceptional cases where the regular process of adding endpoint attributes can not be applied due to time constraints. Usage of these reserve attributes must be explicitly approved by development management before proceeding.	Boolean	True, False	false

Table 105 Create Endpoint Input Fields (Sheet 8 of 12)

Field Name	Description	Length	Range	Default Value
(Attributes) CFLoopExtended		Boolean	True, False	
(Attributes) Reserved5	See Reserved4	Boolean	True, False	false
(Attributes) Reserved6	See Reserved4	Boolean	True, False	false
(Attributes) VideoCallAllowed		Boolean	True, False	
TrafficIsPublicOffNet	Separate endpoints will be provisioned to segregate SIP NNI PSTN/off-net traffic from on-net traffic. The flag is only applicable to SIP NNIs. The default of the flag is false; i.e., not set.	Boolean	false, true	false
SipPrivateNetwork	A new endpoint attribute for SIP private networking, defined by UCE/UCE services. SOAP will support this newly introduced attribute which will process originating and or terminating calls over that signaling interface.			
(Attributes) AllowProxyByPass	Indicates if the Proxy will be bypassed	Boolean	TrueFalse	False
(Attributes) AccbSupported	Indicates whether the ACCB attribute is supported or not	Boolean	TrueFalse	False
(Attributes) DoNotAuditEndpoint	Enable/Disable Audit Mechanism	Boolean	True, False	False
(Attributes) AnatSetting	Enable/Disable Anat Setting	Boolean	True, False	False
(Attributes) SupportCallbackPathReservation	Enable/Disable CallPath reservation	Boolean	True, False	False
(Attributes) SendInternationalNumberInGNF;	New name for the SendUriInTelephoneSubscriberFormat attribute.	Boolean	True, False	false
(Attributes) UseEndpointDefaultHomeDnAsAuthNumber	When enabled, the endpoint's default home DN is selected as authentication number when sending the SIP INVITE via the SIP NNI endpoint sending the SIP INVITE via the SIP NNI endpoint.	Boolean	True, False	false

Table 105 Create Endpoint Input Fields (Sheet 9 of 12)

Field Name	Description	Length	Range	Default Value
(Attributes) UseSubscriberHomeDnAsAuthNumber	When enabled, the subscriber's home DN is selected as authentication number when sending the SIP INVITE via the SIP NNI endpoint.	Boolean	True, False	false
(Attributes) DisableLongCallAudit	When enabled, the long call duration timer is disabled.	Boolean	True, False	false
(Attributes) SendReceiveImpactLevel	If the attribute is false, then no 'Impact Level' notifications shall be sent to this endpoint and any 'Impact Level' notifications received from this endpoint shall be ignored. **WSDL V7.00.15.01**	Boolean	True, False	False [False, True]
(Attributes) ICEEnabled	Indicates whether the endpoint supports ICE	enum	ICEEAutomatic ICEEEnabled	ICEEnabled
EnablePrack	To provide support of the SIP PRACK method. The first step provides a limited support of PRACK on a half-call basis for the OSV as a B2BUA evolving to a future step when end-to-end support can be provided. The first step limited PRACK implementation referenced herein as identified as "PRACK-Lite". **Supported from WSDL V5.00.40.01**	bool	True,False	false
(Attributes) TlsRfc5626Ping	TLS per RFC5626 ping	bool		
(Attributes TlsDualPathMethod	TLS Dual Path Method	bool		
(Attributes) UseEndpointDefaultHomeDnAsAuthNumber	Use the Endpoint Default HomeDn As Authorization Number	bool		
(Attributes)UseSubscriberHomeDnAsAuthNumber	Use Subscriber HomeDn As Authorization Number	bool		
AddressOf Record	The address used by SIP at registration of the endpoint.	0..64	character string	""

Table 105 Create Endpoint Input Fields (Sheet 10 of 12)

Field Name	Description	Length	Range	Default Value
SendPreferredIdentityHeader	If this endpoint attribute is set then the preferred identity header is sent.	Boolean	True False	False
SendDomainNameInFromHeader	If this endpoint attribute is set the host part of the From header of the initial INVITE to an Italtel SIP trunk will contain the OpenScape Voice domain name instead of the IP address.	Boolean	True False	False
SendForwardingNumberInHeaders	If this endpoint attribute is set the forwarding number will be send in headers.	Boolean	True False	False
DoNotSendDiversionHeader	If this endpoint attribute is set then the diversion header will not be sent.	Boolean	True False	False
DoNotSendInviteWithoutSDP	If this endpoint attribute is set then it will be not allowed to send an invite without SDP.	Boolean	True False	False
EnhancedSubscriberRerouting	Enhanced Subscriber Rerouting shall be attempted for calls to an endpoint that is provisioned with the new "Enhanced Subscriber Rerouting" attribute if the last route in the route set towards the endpoint is CAC restricted.This endpoint attribute only applies to non-subscriber endpoints.Note: this flag is intended to be set for Voice Mail Servers.	Boolean	TrueFalse	False
ServerFailover	Indicates whether network server failover is enabled for this endpoint.	enum	EpFailoverOff EpFailoverOn	
KeysetUse	Indicates whether this subscriber is associated with a keyset device, either as a primary or phantom line.	enum	KeysetNone KeysetPrimaryLine KeysetPhantomLine	
NumberAliases	Number of aliases on list	int		
NodeName	Symbolic identifier of endpoint that “owns” the alias or suffix.	str		
AliasName	Name of an alias or alias suffix.	str		
(AliasCreateObject) AliasType	Enumerated value that represents the type of the alias.	int		AtSipUrl
(AliasCreateObject) CustomerName	Symbolic name of the associated customer.	str		

Table 105 Create Endpoint Input Fields (Sheet 11 of 12)

Field Name	Description	Length	Range	Default Value
(AliasCreate Object) Registration	Symbolic name of the associated customer.	int		
EndpointLocation Name	The name of the endpoint location	Character String	1…20	N/A
EndpointLocation Code	The code of the endpoint location	Character String	1…8	N/A
EndpointNNIType	The Type of the NNIType	enum	NNITypeUnused, NNITypeUnassigned, NNITypeOpenScapeVoiceNNIType, Hipath4000, NNITypePSTNGateway, NNITypeOther	NNITypeOpenScapeVoice
EndpointProvider Name	The name of the Endpoint Provider. Note that in order to set the provider the NNIType should be set to PSTNGateway	Character String	1…12	N/A
DnsSrv	The type of the DnsSRV	enum	DnsSrvUnassigned DnsSrvDisabled DnsSrvEnabled	DnsSrvDisabled

Table 105 Create Endpoint Input Fields (Sheet 12 of 12)

8.3.2 Modify Endpoint

This method adds, deletes, and updates an endpoint. Table 106 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
V5.00.18.01		subRC_APP_DSA_LOCATION_DOMAIN_NAME_SIP_PROXY_INVALID
10.00.01	N/A	subRC_SUCCESS

Table 106 Modify Endpoint Return Codes

Table 107 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
Name	CLI name of the endpoint.	str		

Table 107 Modify Endpoint Input Fields (Sheet 1 of 12)

Field Name	Description	Length	Range	Default Value
IpAddress	Character string with the IP Address of the endpoint. If the IP address and port are provided with non-default value, it must be unique in the set of addresses that include: {ipAddress, port} combinations for endpoint {rasIpAddress,rasPort} combinations for endpoint {ipAddress, port} combinations for gatekeepers	str		“”
NewName	New CLI name of the endpoint	str		
AltIpAddress	Optional secondary IP address of the endpoint. The same uniqueness rule (as above in IpAddress) applies in AltIpAddress.	str		
RasIpAddress	IP address of the RAS channel for the SIP endpoint.	str		
ManagementIpAddress	Character string for a separate address for the endpoint allowing its management using a different IP address for its configuration than the signaling address (IpAddress field).The ManagementIpAddress can be provisioned in IPv4, IPv6 or FQDN format. If this field is not configured, the IpAddress of the endpoint is considered to be the management IP address as well.	str		“”
Register Process	RTP name of the process that sends an RRQ request.			
Port	Port is the signaling address of the endpoint. If the IP address and port are provided with non-default value, it must be unique in the set of addresses that include: {ipAddress, port} combinations for endpoint {rasIpAddress,rasPort} combinations for endpoint {ipAddress, port} combinations for gatekeepers	int		0
AltPort	Optional secondary IP address of the endpoint.	int		

Table 107 Modify Endpoint Input Fields (Sheet 2 of 12)

Field Name	Description	Length	Range	Default Value
RasPort	RasPort is an integer that represents the RAS port number at the endpoint.	int		
EndPointType	Enumerated type that represents the type of the endpoint.	int	EtUnassigned EtSip	
Registration	An integer that represents bitwise assignments of five attributes: Operational state of the endpoint (highest 2 bits): UNNASIGNED, ACTIVE, BLOCKED_NEW, BLOCKED_ALL Administrative state of the dynamic endpoint (next 2 bits): UNNASIGNED, ACTIVE, BLOCKED_NEW, BLOCKED_ALL Authentication pre-granted: indicates whether authentication is pregranted for a dynamic endpoint (next 3 bits): UNNASIGNED, NEITHER_SIDE, A_SIDE, B_SIDE, BOTH_SIDES RRQ state: indicates whether a dynamic endpoint is (URQ, RRQ) (next 2 bits): UNNASIGNED, UNREGISTERED, REGISTERED RRQ type: indicates whether the endpoint is static or dynamic (lowest 2 bits): UNNASIGNED, STATIC, DYNAMIC_PORT (static IP), DYNAMIC	int	Default values: **Operstate**: UNNASIGNED=0 **adminState:** ACTIVE=1 **authGranted:** NEITHER_SIDE=1 If the endpoint is static: **rrqState:** UN-REGISTERED=1 **rrqType:** STATIC=1 If the endpoint is dynamic: **rrqState:** REGISTERED=2 **rrqType:** DYNAMIC =3	
TimeToLive	Expiration time for the endpoint (if dynamic).	int		
RrqId	Internal numeric identifier used by the H.323 process to identify the registered endpoint.	Int		
AccountAuth	An enumeration of values for endpoint account authorization. Notes: This field is not modifiable after creation. ccountAuthBasedOnSub applies only to subscriber type endpoints.	enum	AccountAuthUnused, AccountAuthUnassigned, AccountAuthBasedOnSub, AccountAuthBasedOnEp, AccountAuthNoServices	AccountAuthUnused
ProfileId	Numeric identifier of the associated endpoint profile object.	int		
ProfileName	Symbolic name of the association endpoint profile object.	str		

Table 107 Modify Endpoint Input Fields (Sheet 3 of 12)

Field Name	Description	Length	Range	Default Value
Sessions	Maximum number of sessions allowed for the endpoint. Note that for non-PBX subscriber endpoints, this represents the number of messaging sessions, not the number of voice (audio) sessions.	Int	For SIP endpoint, it has a default value 3. For non-SIP endpoint, it takes the default value 1. Otherwise, it must take a value greater than 0 and less than a maximum that is provided as an RTP configuration variable. Default and maximum values are determined by the RTP parameters: Srx/Sip/ default_number_of_sessions and Srx/Sip/max_number_of_sessions.	3
Transport	Transport specifies the transport protocol for a SIP endpoint	int	One of TtUdp, TtTcp, and TtTls	
SdeSrtpSupport	"Best-Effort SRTP with SDES" attribute to determine whether to include the SRTP media in the SDP offer or not.	Enum	SrtpUnassigned SrtpAutomatic SrtpEnabled SrtpDisabled	SrtpDisabled
AnatSupport	Indicates whether the endpoint supports ANAT. AnatOn indicates that the endpoint supports ANAT. AnatOff means that the endpoint does not support ANAT. AnatAuto means whether the endpoint supports ANAT shall be determined by the SIP registrar during registration of the endpoint. AnatUnassigned indicates that the default value should be assigned,	Enum	AnatUnassigned AnatAuto AnatOn, AnatOff	If the registration of the endpoint is static: AnatOn If the registration of the endpoint is dynamic: AnatAuto (For registration types please refer to Endpoint Registration field)
Attributes	A list of the following Boolean flags:			
(Attributes) SendAuthNumAsPAI	Indicates whether the AuthNum is send as PAI	boolean	false true	false
(Attributes) CallingPartyCategory	Indicates whether for this endpoint, the incoming CallingPartyCategory (CPC) is interpreted as the originating class of service.	boolean	false true	false
(Attributes) UpdateOnConfirmedDialaouge	Indicates whether this endpoint uses UPDATE for confirm dialogue support. If false, only SIP:RE-INVITE is used for confirm dialogue support.	boolean	false true	false

Table 107 Modify Endpoint Input Fields (Sheet 4 of 12)

Field Name	Description	Length	Range	Default Value
(Attributes) SendProvResponseOnUpdate	Indicates whether the endpoint sends a provisional response during session updates.	boolean	false true	false
(Attributes) QSigOverSip	Indicates whether this endpoint supports SIPQ signaling.	boolean	false true	false
(Attributes) ImsFor3GPP	Indicates whether the endpoint supports 3GPP IMS. If switch application type is set to IMS MGCP then SIPSM constructs RequestURI in TEL URI format, and TO, FROM and P-Asserted headers host part are constructed with IMS domain name. Note that this attribute is not associated with a SIP subscriber, but with the SIP endpoint, so it can only be set (including being exported and imported) using the Endpoint provisioning requests, not the Subscriber requests.	boolean	false true	false
(Attributes) MTPEnabled	Indicates that the endpoint has precondition signaling enabled.	boolean	false true	false
(Attributes) EarlySessionMedia	Indicates whether the endpoint has early-session media signaling enabled.	boolean	false true	false
(Attributes) TrustedSubscriber	Indicates whether the subscriber is located behind a trusted endpoint.	boolean	false true	false
(Attributes) GSMSubscribe	Indicates whether the endpoint is a GSM Mobile IP Centrex subscriber.	boolean	false true	false
(Attributes) SurvivableProxy	Indicates whether this endpoint is a survivable SIP proxy endpoint.	boolean	false true	false
(Attributes) URISupportRequired	Indicates whether TEL URI must be used for sending addresses on this interface.	boolean	false true	false
(Attributes) URISupportAllowed	Indicates whether TEL URI is allowed to be used for sending addresses on this interface.	boolean	false true	false
(Attributes) SendUriAsReceived	Indicates whether TEL URI address specification is sent as originally received.	boolean	false true	false
(Attributes) SendUriInTelephoneSubscriberFormat	Indicates whether the TEL URI is sent in telephone subscriber (global number or local number) format (including phone context for local numbers).	boolean	false true	false

Table 107 Modify Endpoint Input Fields (Sheet 5 of 12)

Field Name	Description	Length	Range	Default Value
(Attributes) SendUriWithoutPhoneContext	Indicates whether the TEL URI is send in local number format without phone context. Note that the default is true, and this is the only currently supported option for local number format.	boolean	false true	true
(Attributes) SendUriAsDialString	Indicates whether the TEL URI is send in local number format with phone context. Note that the default is false, and this option is not currently supported.	boolean	false true	false
(Attributes) NumericUserIP	Indicates whether or not to treat all numeric user info (without user=phone) as user=ip.	boolean	false true	false
(Attributes) TruncateMIME	Indicates whether the size of the MIME will truncated to reduce its size.	boolean	false true	false
(Attributes) NationalOrInterNationalPrefix	Indicates whether national or international prefix should be inserted.	boolean	false true	false
(Attributes) PDCSBillingInfoSupported	Indicates whether the sending of the STP P-DCS-Billing-Info header parameters is supported.	boolean	false true	false
(Attributes) GenerateReasonHeader	Indicates whether for this endpoint a reason header is generated in response for responses/release messages without a reason header.	boolean	false true	false
(Attributes) SendReasonHeaderForQ850	Indicates whether for this endpoint a reason header is sent for Q.850 protocol - non-2XX final responses mapped to Q.850 protocol specifications are allowed to be sent.	boolean	false true	false
(Attributes) SendReasonHeaderForSip	Indicates whether for this endpoint a reason header is sent for SIP protocol - non-2XX final responses which are SIP-specific (without Q.850 protocol specifications) are allowed to be sent.	boolean	false true	false
(Attributes) DoNotSendReasonHeader	Indicates whether for this endpoint a reason header is not to be sent - provides backward compatibility by filtering the reason header from any response sent to a SIP UA.	boolean	false true	false

Table 107 Modify Endpoint Input Fields (Sheet 6 of 12)

Field Name	Description	Length	Range	Default Value
(Attributes) SipOliRequired	Indicates whether for this endpoint the OLI is to be sent. The format of the OLI to be sent depends on the IsupOliRequired attribute. When WSDL is V5.00.22.01 or less this value is created/ modified/displayed. When WSDL is greater than V5.00.22.01, it is created/ modified/displayed as VoiceMail	boolean	false true	false (Supported until V5.00.22.01)
(Attributes) IsupOliRequired	Indicates whether the OLI is to be signaled as an ISUP_OLI parameter or as an OLI parameter.	boolean	false true	false
(Attributes) IgnoreIngressCpcOli	Indicates whether for this endpoint, the CPC, OLI, or ISUP OLI is ignored if received in the incoming SIP:INVITE.	boolean	false true	false
(Attributes) IgnoreAnswerForAnnouncement	Indicates whether the Answer for announcement should be ignored	Boolen	false true	false
(Attributes) EndpointIsSecure	Indicates whether security is enabled for this endpoint.	boolean	false true	false
(Attributes) EndpointIsProxy	Indicates whether this endpoint is a SIP proxy server. Note that this attribute applies only to non-subscriber endpoints.	boolean	false true	false
(Attributes) RouteViaProxy	Indicates whether all SIP:INVITE messages to SIP UAs that are registered via an associated SIP endpoint are sent to the associated SIP endpoint. Note that this attribute may be true only if EndpointIsProxy is also true.	boolean	false true	false
(Attributes) EndpointIsInaccessible	Indicates whether this endpoint is in an inaccessible state. Note that this attribute is read-only and is only listed here for completeness.	boolean	false true	false
(Attributes) MediaServerRedirectionAllowed	Indicates whether media redirection is allowed for this endpoint.	boolean	false true	false

Table 107 Modify Endpoint Input Fields (Sheet 7 of 12)

Field Name	Description	Length	Range	Default Value
(Attributes) RerouteToAlternateIp	Indicates whether, in the scenario where the first destination is not reachable, the switch will look for an alternate destination for the new request.	boolean	false true	false
(Attributes) UseServerVirtualIp	Indicates whether to use the server virtual IP address for SIP gateways, SIP application server, and SIP voice mail. If TLS is selected as transport type then server to server communication is through mutual authentication.	boolean	false true	false
(Attributes) StaticIMSSub	Indicates whether this is the endpoint of a static IMS subscriber.	boolean	false true	false
(Attributes)CallCenterApplication	Indicates that the endpoint is a call center application.	boolean	false true	false
(Attributes)EndpointIsAssociatedServer	A SIP NNI endpoint assigned with this attribute set to true shall be excluded from Trunking Session licensing scenarios.Note that this attribute applies only in the Enterprise environment.	boolean	false true	false
(Attributes)SendInsecureRefferedByHeader	When a Referred-By header field is included in the initial request sent to the transfer target (for a blind transfer), the Referred-By header (including any generic-parameters) will be passed transparently from the transferor to the transfer target.	boolean	false true	true
(Attributes)SendPubNrInFQNInFromHeader		boolean	false true	
(Attributes) OverrideIRMCodecRestrictions	This attribute forces selection of codecs by overriding IRM restrictions.	Boolean	false true	false
(Attributes) ReroutingSiptoSip	This attribute indicates whether the subscriber rerouting of incoming calls through the SIP endpoint to a survivable SIP subscriber is allowed.	Boolean	false true	false

Table 107 Modify Endpoint Input Fields (Sheet 8 of 12)

Field Name	Description	Length	Range	Default Value
(Attributes) ReroutingFwd	This attribute indicates whether subscriber rerouting of incoming calls through the SIP endpoint that are forwarded to a survivable SIP subscriber is allowed.	Boolean	false true	false
(Attributes) TransferHandoff	If this attribute is set, calls can be transferred to the Genesys.	Boolean	false, true	false
TrafficIsPublicOffNet	Separate endpoints will be provisioned to segregate SIP NNI PSTN/off-net traffic from on-net traffic. The flag is only applicable to SIP NNIs. The default of the flag is false; i.e., not set.	Boolean	false, true	false
SipPrivateNetwork	A new endpoint attribute for SIP private networking, defined by UCE/UCE services. SOAP will support this newly introduced attribute which will process originating and or terminating calls over that signaling interface.			
(Attributes) AllowProxyByPass	Indicates if the Proxy will be bypassed	Boolean	TrueFalse	False
(Attributes) AccbSupported	Indicates whether the ACCB attribute is supported or not	Boolean	TrueFalse	False
(Attributes) DoNotAuditEndpoint	Enable/Disable Audit Mechanism	Boolean	True, False	False
(Attributes) AnatSetting	Enable/Disable Anat Setting	Boolean	True, False	False
(Attributes) SupportCallbackPathReservation	Enable/Disable CallPath reservation	Boolean	True, False	False
(Attributes) SendInternationalNumberInGNF;	New name for the SendUriInTelephoneSubscriberFormat attribute.	Boolean	True, False	false
(Attributes) UseEndpointDefaultHomeDnAsAuthNumber	When enabled, the endpoint's default home DN is selected as authentication number when sending the SIP INVITE via the SIP NNI endpoint sending the SIP INVITE via the SIP NNI endpoint.	Boolean	True, False	false
(Attributes) UseSubscriberHomeDnAsAuthNumber	When enabled, the subscriber's home DN is selected as authentication number when sending the SIP INVITE via the SIP NNI endpoint.	Boolean	True, False	false

Table 107 Modify Endpoint Input Fields (Sheet 9 of 12)

Field Name	Description	Length	Range	Default Value
(Attributes) DisableLongCallAudit	When enabled, the long call duration timer is disabled.	Boolean	True, False	false
SendReceiveImpactLevel	If the attribute is false, then no 'Impact Level' notifications shall be sent to this endpoint and any 'Impact Level' notifications received from this endpoint shall be ignored. **WSDL V7.00.15.01**	No	Boolean	False [False, True]
(Attributes) Reserved4	The purpose for this is to be prepared for future urgent requests requiring a new endpoint attribute and where new functionality via 8K (e)patch is delivered but there is no time to deliver the corresponding Assistant changes at the same time. The intention is to use these reserve attributes only in exceptional cases where the regular process of adding endpoint attributes can not be applied due to time constraints. Usage of these reserve attributes must be explicitly approved by development management before proceeding.	Boolean	True, False	false
(Attributes) CFLoopExtended		Boolean	True, False	
(Attributes) Reserved5	See Reserved4	Boolean	True, False	false
(Attributes) Reserved6	See Reserved4	Boolean	True, False	false
(Attributes) VideoCallAllowed		Boolean	True, False	

Table 107 Modify Endpoint Input Fields (Sheet 10 of 12)

Field Name	Description	Length	Range	Default Value
(Attributes EnablePrack	To provide support of the SIP PRACK method. The first step provides a limited support of PRACK on a half-call basis for the OSV as a B2BUA evolving to a future step when end-to-end support can be provided. The first step limited PRACK implementation referenced herein as identified as “PRACK-Lite”. **Supported from WSDL V5.00.40.01**	bool	True,False	false
(Attributes) TlsRfc5626Ping	TLS per RFC5626 ping	bool		
(Attributes TlsDualPathMethod	TLS Dual Path Method	bool		
(Attributes) UseEndpointDefaultHomeDnAsAuthNumber	Use the Endpoint Default HomeDn As Authorization Number	bool		
(Attributes)UseSubscriberHomeDnAsAuthNumber	Use Subscriber HomeDn As Authorization Number	bool		
(Attributes)SendReceiveImpactLevel	Send Receive Impact Level	bool		
SendPPreferredIdentityHeader	If this endpoint attribute is set then the preferred identity header is sent.	Boolean	True False	False
SendDomainNameInFromHeader	If this endpoint attribute is set the host part of the From header of the initial INVITE to an Italtel SIP trunk will contain the OpenScape Voice domain name instead of the IP address.	Boolean	True False	False
SendForwardingNumberInHeaders	If this endpoint attribute is set the forwarding number will be sent in headers.	Boolean	True False	False
DoNotSendDiversionHeader	If this endpoint attribute is set then the diversion header will not be sent.	Boolean	True False	False
DoNotSendInviteWithoutSDP	If this endpoint attribute is set then it will be not allowed to send an invite without SDP.	Boolean	True False	False

Table 107 Modify Endpoint Input Fields (Sheet 11 of 12)

Field Name	Description	Length	Range	Default Value
EnhancedSubscriberRerouting	Enhanced Subscriber Rerouting shall be attempted for calls to an endpoint that is provisioned with the new "Enhanced Subscriber Rerouting" attribute if the last route in the route set towards the endpoint is CAC restricted.This endpoint attribute only applies to non-subscriber endpoints.Note: this flag is intended to be set for Voice Mail Servers.	Boolean	TrueFalse	False
AddressOf Record	The address used by SIP at registration of the endpoint.	0..64	character string	“”
ServerFailover	Indicates whether network server failover is enabled for this endpoint.	enum	EpFailoverOff EpFailoverOn	
KeysetUse	Indicates whether this subscriber is associated with a keyset device, either as a primary or phantom line.	enum	KeysetNone KeysetPrimaryLine KeysetPhantomLine	
NumberAliases	Number of aliases on list	int		
NodeName	Symbolic identifier of endpoint that “owns” the alias or suffix.	str		
AliasName	Name of an alias or alias suffix.	str		
AliasType	Enumerated value that represents the type of the alias.	int	Fill with AtE164	
EndpointLocation Name	The name of the endpoint location	Character String	1…20	N/A
EndpointLocation Code	The code of the endpoint location	Character String	1…8	N/A
EndpointNNIType	The Type of the NNIType	enum	NNITypeUnused, NNITypeUnassigned, NNITypeOpenScapeVoiceNNIType, Hipath4000, NNITypePSTNGateway, NNITypeOther	NNITypeOpenScapeVoice
EndpointProvider Name	The name of the Endpoint Provider. Note that in order to set the provider the NNIType should be set to PSTNGateway	Character String	1…12	N/A
DnsSrv	The type of the DnsSRV	enum	DnsSrvUnassigned DnsSrvDisabled DnsSrvEnabled	DnsSrvDisabled

Table 107 Modify Endpoint Input Fields (Sheet 12 of 12)

8.3.3 Delete Endpoint

This method deletes an endpoint. Table 108 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01	N/A	subRC_SUCCESS

Table 108 Delete Endpoint Return Codes

Table 109 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
EndpointId	Unique identifier associated with an endpoint in the database. Only this specific endpoint with this id will be removed.	int		
EndpointType	Type of endpoint to be removed.	int	EtUnassigned EtH323Gw EtH323Mcu EtSip EtH323All	
Number Endpoints	Number of endpoint names on list to remove.	int		
Id	Unique numeric identifier of the endpoint. Refers to an existing endpoint in the database. The identifiers can not be modified.	int		
Name	CLI name of the endpoint that will be removed.	str		

Table 109 Delete Endpoint Input Fields

8.3.4 Display Endpoint

This method retrieves the endpoint and displays its parameters. Table 110 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01	N/A	subRC_SUCCESS

Table 110 Display Endpoint Return Codes

Table 111 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
Specific	Refer to the paragraph at the beginning of Chapter 8.			

Table 111 Display Endpoint Input Fields (Sheet 1 of 3)

Field Name	Description	Length	Range	Default Value
Paging	Standard Paging structure for queries. For details, refer to Section 5.5, "Paging", on page 72.			Null
EndpointId	Numeric identifier associated with an endpoint in the database. Used to retrieve the specific endpoint. If EndpointId or EndpointType is unspecified, then all endpoints are retrieved.	int		
EndpointType	The type of the endpoint to retrieve. If EndpointId or EndpointType is unspecified, then all endpoints are retrieved.	int	EtUnassigned EtH323Gw EtH323Mcu EtSip EtH323All	
NumberOf Endpoints	Number of endpoints to be retrieved.	int		
item	Endpoint itself.			
EndpointIsAssociatedServer	A filter item to return only those endpoints which have been defined as associated servers. These endpoints are excluded from trunking session licensing. If not passed (or false), all endpoints are returned.	boolean	false true	false
QueryEndpointType	Modifies the search type • QueryEPTypeAssociatedEndpoints - Query will return the list of endpoints associated to the EpName provided. • QueryEPTypeCandidateRepEndpoints - Used in combination with the BgId tag in order to retrieve the list of endpoints that can be used as Representative in a new Branch Office.	enum	QueryEPTypeDefault QueryEPTypeAssociatedEndpoints QueryEPTypeCandidateRepEndpoints	QueryEPType Default
BGId	The Business Group Index of the BGName we wish to retrieve the candidates for BO Representation	integer		N/A
SortType	Provides a means of sorting the results based on a number of endpoint characteristics	enum	EndpointStUnassigned EndpointStID EndpointStEpType EndpointStAccountAuth EndpointStEpName	EndpointStID
SortDirection	Determines the way the objectsare sorted.	SortOrderAscending - Objects are sorted in ascending order. SortOrderDescending - Objects are sorted in descending order.		SortOrderAscending

Table 111 Display Endpoint Input Fields (Sheet 2 of 3)

Field Name	Description	Length	Range	Default Value
AccountAuthentication	An enumeration of values forendpoint account authorization.	enum	AccountAuthUnused, AccountAuthUnassigned, AccountAuthBasedOnSub, AccountAuthBasedOnEp, AccountAuthNoServices	AccountAuthBasedOnEp
CSTADeviceId	The NID in Device Number format/ Read Only WSDL version V7.00.01.10	int	NA	NA

Table 111 Display Endpoint Input Fields (Sheet 3 of 3)

8.4 Destination

An E164 (or generic destination) is an object in the E.164 domain that conceptually represents the destination of some particular route. A destination is linked to a number of route lists, each of which maintains a list of routes representing possible routes from the softswitch to that destination.

8.4.1 Create Destination

The CreateDestination command allows us to provision one Generic Destination and optionally one associated RouteList with one or more routes.

In order to provision a Generic Destination then we must not pass a number of RouteList and number of routes. The GenericDestination may uniquely be identified by the Destination id (unique numeric identifier generated by the database) and the DestinationName. The fields SignalingType and BearerCap have to take values <= Unassigned.

From the CreateDestination menu we can optionally provision a RouteList to a GenericDestination. The RouteList may uniquely be identified by the E164Destination, the Bearer capability and the signaling type. The Bearer Cap and SignalingType may only take values greater than or equal to UNDEFINED.

Table 112 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01	N/A	subRC_SUCCESS

Table 112 Create Destination Return Codes

Table 113 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
Id	Numeric identifier of the generic destination	Int		
DestinationName	Symbolic name of the destination, of length less than or equal to 15. The pair {Name, NumberPlan) is unique in the table of generic destinations.	str		
DestinationSignaling	Indicates the type of destination signaling. Basically when creating a generic destination, the signaling type should be of value SigUnused. Moreover, the SignalingType does not characterize a generic destination.	enum	SigIgnored SigUnused SigUnassigned SigIsup SigPri, SigMgcp, SigH323, SigSip, SigSipt, SigCas, SigSipQ	SigUnused
Bearer	Assigned bearer capability of the route.	enum		
Destination Attributes			DaNotMSDest DaDest	
Attributes			SigSip SigSipt	
NumberPlanId	Numeric identifier of the numbering plan.	int		
NumberPlan Name	Name of the numbering plan.	str		
NumberRoutes	Number of generic routes	int		
UniqueId	The unique database-created identifier associated with this route in the database.	int		
ReferenceId	Numeric identifier of the referenced object.	int		
Reference Number				
Reference Name	Symbolic name of the referenced object.	str		

Table 113 Create Destination Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
Reference Type	Type of the referenced object.	enum	RtGateway: The referenced endpoint must have the type H323GW RtSipEndpoint: The referenced endpoint must have the type SIP RtTrunk: The referenceId refers to the uniqueIndex of the trunk group. RtGatekeeper RtPri RtMediaGw: if and only if the destination has MSDEST attributes set.	
DigitsToDelete	Number of digits to delete.	int	Takes values between 0 and 15.	0
DigitsToInsert	Optional digit string with characters in the set (0-9,*,#).	6	(0-9,*,#)	EmptyStr
NatureOf Address	Resulting nature of address (enumerated value).	enum	UNDEFINED INTL NATL SUBS NO_DN NW	UNDEFINED

Table 113 Create Destination Input Fields (Sheet 2 of 2)

8.4.2 Modify Destination

This method adds, deletes, and modifies a destination. Table 114 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01	N/A	subRC_SUCCESS

Table 114 Modify Destination Return Codes

Table 115 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
Id	Numeric identifier of the generic destination to modify.	int		
Name	Symbolic name of the destination, of length less than or equal to 15. The pair {Name, NumberPlan) is unique in the table of generic destinations.	str		

Table 115 Modify Destination Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
NewName	New Symbolic name of the destination, of length less than or equal to 15. The pair {Name, NumberPlan) is unique in the table of generic destinations	str		
Signaling		enum		
Bearer	Assigned bearer capability to the route.	enum	CapSpeech CapAudio CapData64KB CapRateAdapted Cap64KBPreffered	
Destination Attributes			DaNotMSDest DaDest	
Attributes	This toggles the “Prioritized” value in a Destination’s Route list	enum	AtrYes AtrNo	
NumberPlanId	Numeric identifier of the numbering plan.	int		
NumberPlan Name	Name of the numbering plan.	str		

Table 115 Modify Destination Input Fields (Sheet 2 of 2)

8.4.3 Delete Destination

This method deletes the current destination. Table 116 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01	N/A	subRC_SUCCESS

Table 116 Delete Destination Return Codes

Table 117 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
Id	Numeric identifier of the generic destination to remove.	int		
Name	Symbolic name of the destinationThe pair {Name, NumberPlan) is unique in the table of generic destinations.	15		
Signaling				

Table 117 Delete Destination Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
Bearer	Assigned bearer capability to the route	enum	CapSpeech CapAudio CapData64KB CapRateAdapted Cap64KBPreffered	
Destination Attributes			DaNotMSDest DaDest	
Attributes				
NumberPlanId	Numeric identifier of the numbering plan.	Int		
NumberPlan Name	Name of the numbering plan.	Str		

Table 117 Delete Destination Input Fields (Sheet 2 of 2)

8.4.4 Display Destination

This method retrieves and displays a destination and its parameters. Table 118 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01	N/A	subRC_SUCCESS

Table 118 Display Destination Return Codes

Table 119 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
Specific	Refer to the paragraph at the beginning of Chapter 8.			
Paging	Standard paging structure for queries.			Null
SortingType	Defines whether sorting is done by {numberPlanName, destId} or destId alone.		StReference StDestnation	StDestnation
Id	Numeric identifier of the generic destination to query.	int		
Name	Symbolic name of the destination. The pair {Name, NumberPlan) is unique in the table of generic destinations.	15		
Signaling				

Table 119 Display Destination Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
Bearer	Bearer capability assigned to the route.	enum	CapSpeech CapAudio CapData64KB CapRateAdapted Cap64KBPreffered CapUnused	
Destination Attributes			DaNotMSDest DaDest	
Attributes				
NumberPlanId	Numeric identifier of the numbering plan.	int		
NumberPlan Name	Name of the numbering plan.	str		
UniqueId				
ReferenceId	Numeric identifier of the referenced object.	int		
Reference Number				
ReferenceName	Symbolic name of the referenced object.	str		
ReferenceType	Type of the referenced object.	enum	RtGateway: The referenced endpoint must have the type H323GW. RtSipEndpoint: The referenced endpoint must have the type SIP. RtTrunk: The referenceId refers to the uniqueIndex of the trunk group. RtGatekeeper RtPri RtMediaGw: if and only if the destination has MSDEST attributes set.	
DigitsToDelete	Number of digits to delete	int	Takes values between 0 and 15.	
DigitsToInsert	Optional digit string with characters in the set (0-9,*,#).	6	(0-9,*,#)	
NatureOfAddress	Resulting nature of address (enumerated value).	enum	NoaUndefined NoaIntl NoaNatl NoaSubs NoaNoDn NoaNw	

Table 119 Display Destination Input Fields (Sheet 2 of 2)

8.5 Rate Area

8.5.1 Create Rate Area

This method creates a rate area. Table 120 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01	N/A	subRC_SUCCESS

Table 120 Create Rate Area Return Codes

Table 121 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
Id				
GroupId	Numeric identifier of the rate area group (always equal to Group plus 2)	Integer	Greater than or equal to (+1)	Optional
Group	Numeric identifier of the rate area group.	Integer	Greater than or equal to (-1)	Optional
Name	Name of the rate area object		Character String	
NpaNxx	Leftmost digit sequence indicating the area code and the exchange for the rate area.		A digit sequence of length less than 16.	
SIhrName		String	Character String	
PointCode		Integer		
NodeVariant		String	NodeVariantUnassigned NodeVariantNone NodeVariantA7 NodeVariantA7M3UA	
Lata	Associated LATA digits.	0..8	Character String	“”
LocalTollName		String		

Table 121 Create Rate Area Input Fields

8.5.2 Remove Rate Area

This method deletes the current rate area. Table 122 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01	N/A	subRC_SUCCESS

Table 122 Delete Rate Area Return Codes

Table 123 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
Id				
Group	Numeric identifier of the rate area group to remove	integer	Greater than or equal to (-1)	
Name	Name of the rate area object to remove.		Character String	
NpaNxx	Leftmost digit sequence indicating the area code and the exchange for the rate area.		A digit sequence of less than 16 digits	

Table 123 Delete Rate Area Input Fields

8.5.3 Query Rate Area

This method retrieves and displays a rate area. Table 124 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01	N/A	subRC_SUCCESS

Table 124 Display Rate Area Return Codes

Table 125 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
Specific	Refer to the paragraph at the beginning of Chapter 8.			
Paging	Standard paging structure for queries.			Null
Id				
GroupId	Numeric identifier of the rate area group to remove.	integer	Greater than or equal to (+1)	
Group	Numeric identifier of the rate area group to remove	integer	Greater than or equal to (-1)	
Name	Name of the rate area object to remove.		Character String	
NpaNxx	Leftmost digit sequence indicating the area code and the exchange for the rate area.		A digit sequence of less than 16 digits	
SlhrName		String	Character String	
PointCode		Integer		

Table 125 Display Rate Area Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
NodeVariant		String	NodeVariantUnassigned NodeVariantNone NodeVariantA7 NodeVariantA7M3UA	
Lata	Associated LATA digits.	0..8	Character String	“”
LocalTollName		String		
QueryRateAreaType		String	QueryRATypeDefault QueryRAUniqueRateAreaNames	
SortType		String	RateAreaStUnassigned RateAreaStRateAreaName	
SortDirection		String	SortOrderAscending SortOrderDescending	

Table 125 Display Rate Area Input Fields (Sheet 2 of 2)

8.6 Class of Service

8.6.1 Create Class of Service

This method creates a class of service. Table 126 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01	N/A	subRC_SUCCESS

Table 126 Create Class of Service Return Codes

Table 127 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
Name	Symbolic name of the class of service.	str		
Id	Numeric identifier of the class of service.	int		
Number	User-supplied identifier of the class of service.	int	Greater than or equal to (+1)	Optional
DefaultCpcCos		int		
DefaultCpcCosName		string		

Table 127 Create Class of Service Input Fields

8.6.2 Delete Class of Service

This method deletes a class of service. Table 128 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01	N/A	subRC_SUCCESS

Table 128 Delete Class of Service Return Codes

Table 129 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
Name	Symbolic name of the class of service.	str		
Id	Numeric identifier of the class of service.	int		
Number	User-supplied identifier of the class of service.	int	Greater than or equal to (-1)	
DefaultCpcCos				
DefaultCpcCosName				

Table 129 Delete Class of Service Input Fields

8.6.3 Modify Class of Service

This method modifies a class of service. Table 130 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01	N/A	subRC_SUCCESS

Table 130 Modify Class of Service Return Codes

Table 131 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
Name	Symbolic name of the class of service.	str		
NewName	New Symbolic name of the class of Service	str		
Id	Numeric identifier of the class of service.	int		

Table 131 Modify Class of Service Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
Number	User-supplied identifier of the class of service.	int		-2
DefaultCpcCos		int		
DefaultCpcCosName		string		

Table 131 Modify Class of Service Input Fields (Sheet 2 of 2)

8.6.4 Query Class of Service

This method retrieves and displays a class of service. Table 132 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01	N/A	subRC_SUCCESS

Table 132 Display Class of Service Return Codes

Table 133 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
Specific	Refer to the paragraph at the beginning of Chapter 8.			
Paging	Standard paging structure for queries.			Null
Name	Symbolic name of the class of service to query.	Str		
Id	Numeric identifier of the class of service to query.	Int		
Number	User-supplied identifier of the class of service.	int		-2

Table 133 Display Class of Service Input Fields

8.7 Routing

This API adds routes to an existing route list. Otherwise, this command creates a new route list for an existing destination.

8.7.1 Create Routing

This method creates a route. Table 134 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01	N/A	subRC_SUCCESS

Table 134 Create Routing Return Codes

Table 135 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
Operator Id				
Id				
Name				
Signaling	Specifies the A-side signaling type. If signaling type is assigned, the A-side signaling protocol must match the signaling type in order to use the route list. If a signaling type is not assigned, the route list can be used regardless of the A-side signaling protocol. Signaling type SIPQ indicates that the incoming call is from the SIPQ SM and will be tandemed to a route with org_signaling_type=SIPQ.	string	SigIgnored SigUnused SigUnassigned SigIsup SigPri, SigMgcp, SigH323, SigSip, SigSipt, SigCas, SigSipQ	
Bearer	Bearer capability assigned to the route.	enum	CapIgnored CapUnused CapUnassigned CapSpeech CapAudio CapData64KB CapRateAdapted Cap64KB Cap64KBPreffered	
Destination Attributes	Specifies whether the destination is a media server destination.		DaNotMSDest DaDest	
Attributes				
NumberPlanId	Numeric Identifier of the numbering plan.	int		
NumberPlan Name	Name of the numbering plan.	str		
NumRoutes				
UniqueId				
ReferenceId	Numeric identifier of the referenced object	int		
Reference Number				

Table 135 Create Routing Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
Reference Name	Symbolic name of the route referenced object	str		
Reference Type	Specifies an integer variable indicating the type of the route reference object.	enum	GW GK SIPEP	
DigitsToDelete	Number of leading digits to be deleted from the destination DN.	Int	Takes values between 0 and 15.	0
DigitsToInsert	Optional digit string with characters in the set (0,9,*,#). It represents the leading digits to add at the beginning of the destination DN.	6		
NatureOf Address	Specifies the nature of address of the destination DN after digit deletion and/or insertion.	Enum	NoaUndefined NoaIntl NoaNatl NoaSubs NoaNoDn NoaNw	Undefined (For PBX)
CacSignaling				
LocalTollName		String		
DigitsSubscriber		String		
DigitsNational		String		
DigitsInternational		String		
LocalNumbersOnly		Boolean		
isPrefixed		Boolean		

Table 135 Create Routing Input Fields (Sheet 2 of 2)

8.7.2 Prioritize Routing

This method prioritizes routing. Table 136 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01	N/A	subRC_SUCCESS

Table 136 Prioritize Routing Return Codes

Table 137 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.			

Table 137 Prioritize Routing Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
UniqueId				
Name				
DestinationId	Specifies a unique, non reusable index number assigned when a destination is created	int		
Destination Name	Specifies an alphanumeric string that uniquely identifies a Destination.	15	Maximum length is 15 character	
Signaling	Specifies the A-Side Signaling Type. If signaling type is assigned, the A-side signaling protocol must match the signaling type in order to use the route list. If signaling type is not assigned, the route list can be used irrespective of the A-Side signaling protocol.	enum	SigIgnored SigUnused SigUnassigned SigSip SigSipt	
Bearer	Bearer capability assigned to the route.	enum	CapIgnored CapUnused CapUnassigned CapSpeech CapAudio CapData64KB CapRateAdapted Cap64KB Cap64KBPreffered	
RouteIndex	Specifies a unique, non-reusable index assigned when a route is created.			
ReferenceId	Numeric identifier of the referenced object	int		
Reference Number				
Reference Name	Symbolic name of the route referenced object	str		
Reference Type	Specifies an integer variable indicating the type of the route reference object.	enum	GW GK SIPEP	
NewRoute Index				

Table 137 Prioritize Routing Input Fields (Sheet 2 of 2)

8.7.3 Modify Routing

This method modifies routing. Table 138 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01	N/A	subRC_SUCCESS

Table 138 Modify Routing Return Codes

Table 139lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.			
UniqueId				
Name				
Signaling	Specifies the A-Side Signaling Type. If a signaling type is assigned, the A-side signaling protocol must match the signaling type in order to use the route list. If signaling type is not assigned, the route list can be used regardless of the A-Side signaling protocol.	enum	SigIgnored SigUnused SigUnassigned SigSip SigSipt	
Bearer	Bearer capability assigned to the route.	enum	CapIgnored CapUnused CapUnassigned CapSpeech CapAudio CapData64KB CapRateAdapted Cap64KB Cap64KBPreffered	
Destination Attributes	Specifies whether or not the destination is a media server destination.		DaNotMSDest DaDest	
Attributes				
NumberPlanId	Numeric Identifier of the numbering plan.	int		
NumberPlan Name	Name of the numbering plan.	str		
UniqueId				
Id				
Name				
ReferenceId	Numeric identifier of the referenced object	int		
Reference Number				

Table 139 Modify Routing Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
Reference Name	Symbolic name of the route referenced object	str		
Reference Type	Specifies an integer variable indicating the type of the route reference object.	enum	GW GK SIPEP	
DigitsToDelete	Number of leading digits to be deleted from the destination DN.	int	Takes values between 0 and 15	0
DigitsToInsert	Optional digit string that represents the leading digits to add at the beginning of the destination DN.	6	(0,9,*,#)	
NatureOf Address	Specifies the nature of address of the destination DN after digit deletion and/or insertion.	Enum	NoaUndefined NoaIntl NoaNatl NoaSubs NoaNoDn NoaNw	Undefined (For PBX)
CacSignaling				
LocalTollName		String		
DigitsSubscriber		String		
DigitsNational		String		
DigitsInternational		String		
LocalNumbersOnly		Boolean		
isPrefixed		Boolean		

Table 139 Modify Routing Input Fields (Sheet 2 of 2)

8.7.4 Query Routing

This method queries routing. Table 140 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01	N/A	subRC_SUCCESS

Table 140 Query Routing Return Codes

Table 141 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.			

Table 141 Query Routing Input Fields (Sheet 1 of 3)

Field Name	Description	Length	Range	Default Value
Specific	Refer to the paragraph at the beginning of Chapter 8.			
PageSize				
PageNumber				
SortType	Defines whether the objects are retrieved with a 2-index or 4-index sort.			
Id				
Name				
Signaling	Specifies the A-side signaling type. If signaling type is assigned, the A-side signaling protocol must match the signaling type in order to use the route list. If a signaling type is not assigned, the route list can be used regardless of the A-side signaling protocol. Signaling type SIPQ indicates that the incoming call is from the SIPQ SM and will be tandemed to a route with org_signaling_type=SIPQ.	string	SigIgnored SigUnused SigUnassigned SigIsup SigPri, SigMgcp, SigH323, SigSip, SigSipt, SigCas, SigSipQ	
Bearer	Bearer capability assigned to the route.	enum	CapIgnored CapUnused CapUnassigned CapSpeech CapAudio CapData64KB CapRateAdapted Cap64KB Cap64KBPreffered	
Destination Attributes	Specifies whether or not the destination is a media server destination.		DaNotMSDest DaDest	
Attributes				
NumberPlanId	Numeric Identifier of the numbering plan.	int		
NumberPlan Name	Name of the numbering plan.	str		
UniqueId				
Id				
Name				
ReferenceId	Numeric identifier of the referenced object	int		
Reference Number				

Table 141 Query Routing Input Fields (Sheet 2 of 3)

Field Name	Description	Length	Range	Default Value
Reference Name	Symbolic name of the route referenced object	str		
Reference Type	Specifies an integer variable indicating the type of the route reference object.	enum	GW GK SIPEP	
DigitsToDelete	Number of leading digits to be deleted from the destination DN.	int	Takes values between 0 and 15	0
DigitsToInsert	Optional digit string that represents the leading digits to add at the beginning of the destination DN.	6	(0,9,*,#)	
NatureOf Address	Specifies the nature of address of the destination DN after digit deletion and/or insertion.	Enum	NoaUndefined NoaIntl NoaNatl NoaSubs NoaNoDn NoaNw	Undefined (For PBX)
CacSignaling				
NumRoutes				
LocalTollName		String		
DigitsSubscriber		String		
DigitsNational		String		
DigitsInternational		String		
LocalNumbersOnly		Boolean		
isPrefixed		Boolean		

Table 141 Query Routing Input Fields (Sheet 3 of 3)

8.7.5 Unassign Routing

This method unassigns a route. Table 142 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01	N/A	subRC_SUCCESS

Table 142 Unassign Routing Return Codes

Table 143 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.			
Id				
Name				
Signaling	Specifies the A-Side Signaling Type. If signaling type is assigned, the A-side signaling protocol must match the signaling type in order to use the route list. If signaling type is not assigned, the route list can be used irrespective of the A-Side signaling protocol.	enum	SigIgnored SigUnused SigUnassigned SigSip SigSipt	
Bearer	Bearer capability assigned to the route.	enum	CapIgnored CapUnused CapUnassigned CapSpeech CapAudio CapData64KB CapRateAdapted Cap64KB Cap64KBPreffered	
Destination Attributes	Specifies whether or not the destination is a media server destination.		DaNotMSDest DaDest	
Attributes				
NumberPlanId	Numeric Identifier of the numbering plan.	int		
NumberPlan Name	Name of the numbering plan.	str		
RouteId	Specifies a user-assigned integer value that uniquely identifies the route in the route list that is associated with the Destination ID/Bearer Capabilities pair. NOTE: To create a destination without a route, the Route ID must be 0.	int	0-255	
NumRoutes				

Table 143 Unassign Routing Input Fields

8.8 Media Server

The OpenScape Voice system supports the RG 8700, the HiPath 4000 and Mediatrix media gateways. Note that these are all SIP endpoints and are therefore not configured using the “Create Media Gateway” SOAP commands. For the OpenScape Voice system, the only Media Gateway that can be created using this command is a Media Server (supported: OpenScape Media Server, Convedia Media Server).

8.8.1 Create Media Server

This method creates a media Server. Table 144 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.02	N/A	subRC_SUCCESS

Table 144 Create Media Server Return Codes

Table 145 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Operator information used for logging.	str	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.	
GatewayName	Name of the media gateway to create	str		
GatewayProtocolType	Type of protocol associated with gateway to be created.	enum	GptUnknown, GptUndefined, GptMgcp, GptSip, GptOther	GptMgcp
GatewayProtocolVersion	Version of the protocol used to communicate with gateway.	enum	GpVerUnused GpVerMgcp0Dot0 GpVerMgcp0Dot1 GpVerMgcp1Dot0NCS1Dot0 GpVerMgcp1Dot0 GpVerMgcp0Dot5 GpVerMgcp1Dot0Tgcp1Dot0 GpVerMegaco1Dot0	GpVerMgcp1Dot0NCS1Dot0
MgListenPort	Media Gateway listen ports.	int array		
MgReceivePort	Media Gateway receive ports.	int array		
MgCircuitFormat	Circuit format supported by the gateway.	str		$/$

Table 145 Create Media Server Input Fields (Sheet 1 of 3)

Field Name	Description	Length	Range	Default Value
NetworkTypes	Information on the type of network on which the gateway is located.	Boolean	{ Internet,Atm,Local }	false
MgAddrCallp	Ip Address of the Media gateway controller (OS Voice ccm).	str		
MgAddrProvision	Ip Address of the Media Gateway.	str		
Timeout	Timeout interval for MGCP requests sent to gateway.	uint		500
RetryCount	Number of times to retry the mgcp requests.	uint		7
IpAddrAllocMetho d	IpAddress allocation method (DNS/Static/Dynamic).	enum	IpAmUndefined IpAmDnsQuery IpAmStatic	IpAmDnsQuery
Fqdn	Fully qualified domain name for the media gateway.	str		
LocalClli	Common language location identifier	str		
GwAssignMethod	Method for choosing the gateway.	enum	GwAmUndefined GwAmAutomatic GwAmNodePrimary GwAmNodeSecondary	GwAmAutomatic
TimerHist	Transaction History timer	int		30
TimerMax	Maximum retransmission timer	int		20
TransmissionTime rLen		int		5
FaxT38Relay	Fax info for gateway	enum	T38FaxUnassigned T38FaxEnabled T38FaxDisabled	T38FaxDisabled
DtmfRelay	DTMF info for the gateway.		DtmfUnassigned DtmfEnabled DtmfDisabled	DtmfDisabled
DtmfRelay	DTMF info for the gateway.	enum	DtmfUnassigned DtmfEnabled DtmfDisabled	DtmfDisabled
MultiHomingFlag	Indicates whether MultiHoming is enabled or not. NOTE: Note that this flag can never be reset by a Modify Media gateway command (it can be set from false to true but not from true to false - in order to do this, the media gateway must be removed and then created again).	enum	MultiHomingEnabled MultiHomingDisabled MultiHomingUndefined	
OverloadSupport	Indicates whether overload support is enabled or not.	enum	YnYes YnNo YnUndefined	YnNo

Table 145 Create Media Server Input Fields (Sheet 2 of 3)

Field Name	Description	Length	Range	Default Value
OverloadAllowCaleaCalls	Not applicable to OpenScape Voice. Indicates whether or not CALEA calls are allowed while in overload.	enum	YnYes YnNo YnUndefined	YnNo
OverloadAllowTwCalls	Indicates whether or not three way calls are allowed while in overload.	enum	YnYes YnNo YnUndefined	YnNo
OverloadReturnCodeOne	Associated Return Code - Level 1	integer	0..9999999999 (seconds)	409
OverloadReturnCodeTwo	Associated Return Code - Level 2	integer	0..9999999999 (seconds)	490
OverloadReturnCodeThree	Associated Return Code - Level 3	integer	0..9999999999 (seconds)	491
OverloadGapIntervalOne	Gap Interval - Level 1	integer	0, 400..499	5
OverloadGapIntervalTwo	Gap Interval - Level 2	integer	0, 400..499	10
OverloadGapIntervalThree	Gap Interval - Level 3	integer	0, 400..499	20
Location Domain	Location Domain name	Char array	[1…64]	NA

Table 145 Create Media Server Input Fields (Sheet 3 of 3)

8.8.2 Modify Media Server

This method modifies a media Server. Table 146 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.02	N/A	subRC_SUCCESS

Table 146 Modify Media Server Return Codes

Table 147 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	Operator information used for logging.	str	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.	
AdminStatus	Gateway Blocked or made active with this field.	enum	RsUnassigned RsAvailable RsBlocked	RsBlocked

Table 147 Modify Media Server Input Fields (Sheet 1 of 3)

Field Name	Description	Length	Range/Values	Default Value
GatewayName	Name of the media gateway to modify	str		
NewGatewayName	New Name of the media gateway	str		
GatewayProtocolType	Type of protocol associated with gateway to be modified.	enum	GptUnknown, GptUndefined, GptMgcp, GptSip,	GptMgcp
GatewayProtocolVersion	Version of the protocol used to communicate with gateway.	enum	GpVerUnused, GpVerMgcp0Dot0, GpVerMgcp0Dot1, GpVerMgcp1Dot0NCS1Dot0 GpVerMgcp1Dot0, GpVerMgcp0Dot5, GpVerMgcp1Dot0Tgcp1Dot0,	Non modifiable attribute.
MgListenPort	Media Gateway listen ports.	int array		
MgReceivePort	Media Gateway receive ports.	int array		
MgCircuitFormat	Circuit format supported by the gateway.	str		
NetworkTypes	Information on the type of network on which the gateway is located.	Boolean	{ Internet,Atm,Local }	false
MgAddrProvision	Ip Address of the Media Gateway.	str		
Timeout	Timeout interval for MGCP requests sent to gateway.	uint		preserved
RetryCount	Number of times to retry the mgcp requests.	uint		preserved
IpAddrAllocMethod	IpAddress allocation menthod(DNS/Static/ Dynamic).	enum	IpAmUndefined IpAmDnsQuery IpAmStatic	preserved
Fqdn	Fully qualified domain name for the media gateway.	str		
LocalClli	Common language location identifier	str		
GwAssignMethod	Method for choosing the gateway.	enum	GwAmUndefined GwAmAutomatic GwAmNodePrimary GwAmNodeSecondary	preserved
TimerHist	Transaction History timer	int		30
TimerMax	Maximum retransmission timer	int		preserved
TransmissionTimerLen		int		preserved

Table 147 Modify Media Server Input Fields (Sheet 2 of 3)

Field Name	Description	Length	Range/Values	Default Value
FaxT38Relay	Fax info for gateway	enum	T38FaxUnassigned T38FaxEnabled T38FaxDisabled	preserved
DtmfRelay	DTMF info for the gateway.		DtmfUnassigned DtmfEnabled DtmfDisabled	preserved
Location Domain	Location Domain name	Char array	[1…64]	NA

Table 147 Modify Media Server Input Fields (Sheet 3 of 3)

8.8.3 Remove Media Server

This method deletes the current media Server. Table 148 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.02	N/A	subRC_SUCCESS

Table 148 Remove Media Server Return Codes

Table 149 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Operator information used for logging.	str	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.	
GatewayName	Name of the media gateway to remove	str		

Table 149 Remove Media Server Input Fields

8.8.4 Block / Unblock Media Server

This method blocks or unblocks a media server.

Table 150 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.02	N/A	subRC_SUCCESS

Table 150 Block / Unblock Media Server Return Codes

Table 151 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Operator information used for logging.	str	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.	
GatewayName	Name of the media gateway to remove	str		
AdminStatus	Admin status of gateway	enum	RsUnassigned, RsAvailable, RsBlocked,	RsAvailable

Table 151 Block / Unblock Media Server Input Fields

8.8.5 Query Media Server

This method retrieves and displays a media server and its attributes. Table 152 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.02	N/A	subRC_SUCCESS

Table 152 Query Media Server Return Codes

Table 153 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Operator information used for logging.	str	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.	
Specific	Indicates type of query. e.g. specific gateway or all gateways,	Boolean		
Paging	Standard paging structure			
GatewayName	Name of the media gateway to remove	str		

Table 153 Query Media Server Input Fields

The response to the Query Media Gateway request contains the attributes listed for the Create Media Gateway request, and in addition, the read-only attributes listed in Table 154:

Field Name	Description	Length	Range	Default Value
MaxNumOfConfPortsPerConf	Maximum number of conference ports per conference.	integer		**3000**

Table 154 Query Media Gateway Attributes (Read Only)

Field Name	Description	Length	Range	Default Value
MaxNumOfConfPortsPerConfForSCC	Maximum number of conference ports per conference for Station Controlled Conference.	integer		**16**
SystemConfPorts	Number of system conference ports existing.	integer		**0**
AvailableConfPorts	Calculated number of available conference ports.	integer		**0**

Table 154 Query Media Gateway Attributes (Read Only)

8.9 Media Server Circuits

8.9.1 Create Media Server Circuits

This method creates media gateway circuits. Table 155 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.02	N/A	subRC_SUCCESS

Table 155 Create Media Server Circuits Return Codes

Table 156 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Operator information used for logging.	str	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.	
GatewayName	Name of the media gateway circuit to create.	str		
CircuitType	Name of the media gateway to remove	enum	CircuitUnused CircuitDoesNotExist CircuitNone CircuitTrunk CircuitPri CircuitLine CircuitMedia CircuitMsAnn CircuitMsSurv CircuitMsConf CircuitMsConfAdhoc CircuitMsAudit	CircuitDoesNot Exist

Table 156 Create Media Server Circuits Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
StartCircuitId	Name of the media gateway to remove	str		
EndCircuitId	Name of the media gateway to remove	str		
OperatorId	Operator information used for logging.	str	Please refer to the definition of Operator ID in Section 5.4.2, "Field Definitions", on page 67.	

Table 156 Create Media Server Circuits Input Fields (Sheet 2 of 2)

8.9.2 Remove Media Server Circuits

This method creates media server circuits. Table 157 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.02	N/A	subRC_SUCCESS

Table 157 Remove Media Server Circuits Return Codes

Table 158 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Operator information used for logging.	str	Please refer to the definition of Operator ID in Section 5.4.2, "Field Definitions", on page 67.	
GatewayName	Name of the media gateway circuit to create	str		
TrunkGroupInfo	Name of the media gateway to remove	enum	TrunkGroupNoForMsUndefined TrunkGroupNoForMsAny TrunkGroupNoForMsAnn TrunkGroupNoForMsSurv TrunkGroupNoForMsConf TrunkGroupNoForMsConfAdhoc TrunkGroupNoForMsAudit	TrunkGroupNoForMsAny
StartCircuitId	Name of the media gateway to remove	str		
EndCircuitId	Name of the media gateway to remove	str		

Table 158 Remove Media Server Circuits Input Fields

8.9.3 Query Media Server Circuits

This method retrieves and displays media server circuits. Table 159 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.02	N/A	subRC_SUCCESS

Table 159 Query Media Server Circuits Return Codes

Table 160 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	Operator information used for logging.	str	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.	
Specific	Indicates type of query. e.g. specific gateway circuits or all gateway circuits	Refer to the paragraph at the beginning of Chapter 8.		
Paging	Standard paging structure			
GatewayName	Name of the media gateway whose circuits are to be displayed.	str		
StartCircuitId	Circuit id to query	str		

Table 160 Query Media Server Circuits Input Fields

Table 161 lists the output fields for this method.

Field Name	Description	Length	Range	Default Value
PaginatedInfo	Standard paging structure. For details, refer to Section 5.5, “Paging”, on page 72.			
MediaGatewayCircuitResultData	An array of MediaGatewayCircuit entry items, each of which contains the following fields:			
GatewayName	Name of the media gateway to which the displayed circuit belongs	str		

Table 161 Query Media Server Circuits Output Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
CircuitType	The type of the circuit being displayed.	enum	CircuitUnused CircuitDoesNotExist CircuitNone CircuitTrunk CircuitPri CircuitLine CircuitMedia CircuitMsAnn CircuitMsSurv CircuitMsConf CircuitMsConfAdhoc CircuitMsAudit	N/A
GatewayType	The type of the gateway to which the circuit belongs.	enum	GwTypeUnkown GwTypeSmx2100 GwTypeMgOther GwTypeHig1100 GwTypeHig1400 GwTypeATA GwTypeUndefined	N/A
StartCircuitId	The ID of the circuit being displayed	str		
EndCircuitId	The ID of the circuit being displayed	str		

Table 161 Query Media Server Circuits Output Fields (Sheet 2 of 2)

8.10 CAC Group

8.10.1 Create CAC Group

This method creates a Call Admission Control (CAC) Group. Table 162 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
	N/A	subRC_SUCCESS subRC_APP_CAC_INVALID_BG_NAME subRC_APP_CAC_INVALID_BO_NAME

Table 162 Create CAC Group Return Codes

Table 163 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
CACGroupName	The name of the CAC Group to be created.	1..31	Character String	required
CACGroupType	Indicates what this CAC Group is based on.	enum	SubNetType, DNAddrType, IpAddrType ParentCACGroups	required
AddressArray	One or more address strings. The valid values for these strings are based on CACGroupType		For CACGroupType=SubNetType values like: 172.1.10.0/24 For CACGroupType=IpAddrType, values like: 10.151.1.10 For CACGroupType=DNAddrTyp e, values like: 15615550110 (a specific directory number) or 1561556* (a directory number prefix) For ParentCACgroup=Add CAC group NAME	(empty array)

Table 163 Create CAC Group Input Fields (Sheet 1 of 3)

Field Name	Description	Length	Range/Values	Default Value
DynamicCACRouterAddress.RouterIpAddress	An IP address string representing the single data center router associated with the CAC group serving the WAN in a collocated nodes scenario or the first router associated with the CAC group serving the WAN in a geo-separated nodes scenario. Presence of the router IP address combined with CACAccessLinkType of the first access link of CACAccessLinkTypeWAN (see below) indicates that the policies associated with this CAC group can be configured with the backup bandwidth values. Note that the RouterIPAddress must be unique and must uniquely identify only one CAC group. If the IP address is already used by a different CAC group, an error is returned.	1..30	Character string	“”
DynamicCACRouterAddress.InterfaceName	Identifies the interface or GRE/IPSec tunnel within the router. The string is of the form “eth0”, “eth1”, etc., if it characterizes an interface name. Note, this parameter is required if the RouterIpAddress is provided. Both can also be omitted.	1..255	Character string	“”
CACAccessLinkType	The first access link type ofthe CAC group	enum	Please refer to the definition of CACGroupName in Section 8.10.1, “Create CAC Group”.	

Table 163 Create CAC Group Input Fields (Sheet 2 of 3)

Field Name	Description	Length	Range/Values	Default Value
DynamicCACRouterAddres s. CACAccessLinkType	The first access link type of the CAC group. Possible values are: • CACAccessLinkTypeWAN - Indicates either a collocated nodes scenario or a geo-separated nodes scenario and a second access link is not valid. • CACAccessLinkTypeDataCe nter - Indicates a collocated nodes scenario and a second access link is not valid. • CACAccessLinkTypeDataCe nter1 - Indicates a geo-separated nodes scenario and a second access link must be defined. Type CACAccessLinkType-DataCenter2 is invalid for the first access link.	enum	CACAccessLinkTypeWAN, CACAccessLinkTypeDataCen ter, CACAccessLinkTypeDataCen ter1, CACAccessLinkTypeDataCen ter2	CACAccessLink TypeWAN
GeoSeparatedDynamicCAC RouterAddress.RouterIPAd dress	An IP address string representing the second data center router associated with the CAC group serving the WAN in the case of geo-separated nodes scenario.	1..30	Character string	“ “
GeoSeparatedDynamicCAC RouterAddress.InterfaceNa me	A string that identifies the tunnel name that links the second data center router with the branch office.	1..255	Character string	“ “
GeoSeparatedDynamicCAC RouterAddress.CACAccess LinkType	The second access link type of the CAC group. The only possible value is: • CACAccessLinkTypeDataCe nter2 - Indicates a geo-separated nodes scenario and a first access link must have been defined. All remaining types are invalid for the second access link.	enum	CACAccessLinkTypeWAN, CACAccessLinkTypeDataCen ter, CACAccessLinkTypeDataCen ter1, CACAccessLinkTypeDataCen ter2	CACAccessLink TypeWAN
BGName	Business Group Name to be associated with CAC Group	Char	0…30	optional
BOName	Branch Office Name to be associated with CAC Group	char	0…30	Optional

Table 163 Create CAC Group Input Fields (Sheet 3 of 3)

8.10.2 Delete CAC Group

This method deletes a Call Admission Control (CAC) Group. Table 164 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
TBD	N/A	subRC_SUCCESS

Table 164 Delete CAC Group Return Codes

Table 165 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
CACGroupName	The name of the CAC Group to be deleted.	Please refer to the definition of CACGroupName in Section 8.10.1, “Create CAC Group”.		required

Table 165 Delete CAC Group Input Fields

8.10.3 Modify CAC Group

This method modifies a Call Admission Control (CAC) Group. Currently only the CACGroupName attribute may be modified. Table 166 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
	N/A	subRC_SUCCESS subRC_APP_CAC_INVALID_BG_NAME subRC_APP_CAC_INVALID_BO_NAME

Table 166 Modify CAC Group Return Codes

Table 167 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
BusinessGroupName	The name of the Business Group associated with the Call Admission Control (CAC) Group	0..30	Character String	“”

Table 167 Modify CAC Group Input Fields

Field Name	Description	Length	Range/Values	Default Value
BranchOfficeName	The name of the Branch Office associated with the Call Admission Control (CAC) Group		Character String	“”
CACGroupName	The name of the CAC Group to be modified.	Please refer to the definition of CACGroupName in Section 8.10.1, “Create CAC Group”.		required
newCACGroupName	The new name for the CAC Group to be modified.	Please refer to the definition of CACGroupName in Section 8.10.1, “Create CAC Group”.		N/A
DynamicCACRouterAddres s. RouterIpAddress	An IP Address string representing the single Data Center router associated with the CAC group serving the WAN in a collocated nodes scenario or the first router associated with the CAC group serving the WAN in a geo-separated nodes scenario. Presence of the Router IP address combined with CACAccessLinkType of the first access link of CACAccessLinkTypeWAN (see below) indicates that the policies associated with this CAC group can be configured with the backup bandwidth values. Note that the RouterIPAddress must be unique and must uniquely identify only one CAC group. If the IP address is already used by a different CAC group, an error is returned.	Please refer to the definition of RouterIpAddress in Section 8.10.1, “Create CAC Group”.		N/A
DynamicCACRouterAddres s. InterfaceName	This identifies the interface or GRE/IPSec tunnel within the router. The string is of the form “eth0”, “eth1”, etc., if it characterizes an interface name.	Please refer to the definition of InterfaceName in Section 8.10.1, “Create CAC Group”.		N/A

Table 167 Modify CAC Group Input Fields

Field Name	Description	Length	Range/Values	Default Value
DynamicCACRouterAddres s. CACAccessLinkType	The first access link type of the CAC group. Possible values are: • CACAccessLinkTypeWAN - Indicates either a collocated nodes scenario or a geo-separated nodes scenario and a second access link is not valid. • CACAccessLinkTypeDataCe nter - Indicates a collocated nodes scenario and a second access link is not valid. • CACAccessLinkTypeDataCe nter1 - Indicates a geo-separated nodes scenario and a second access link must be defined. Type CACAccessLinkType-DataCenter2 is invalid for the first access link.	Please refer to the definition of CACAccessLinkType in Section 8.10.1, “Create CAC Group”.		N/A
GeoSeparatedDynamicCAC RouterAddress.RouterIPAd dress	An IP address string representing the second data center router associated with the CAC group serving the WAN in the case of geo-separated nodes scenario.	Please refer to the definition of RouterIpAddress in Section 8.10.1, “Create CAC Group”.		N/A
GeoSeparatedDynamicCAC RouterAddress.InterfaceNa me	A string that identifies the tunnel name that links the second data center router with the branch office.	Please refer to the definition of InterfaceName in Section 8.10.1, “Create CAC Group”.		N/A
GeoSeparatedDynamicCAC RouterAddress.CACAccess LinkType	The second access link type of the CAC group. The only possible value is: • CACAccessLinkTypeDataCe nter2 - Indicates a geo-separated nodes scenario and a first access link must have been defined. All remaining types are invalid for the second access link.	Please refer to the definition of CACAccessLinkType in Section 8.10.1, “Create CAC Group”.		N/A
BGName	Business Group Name to be associated with CAC Group	Char	0…30	optional
BOName	Branch Office Name to be associated with CAC Group	char	0…30	Optional

Table 167 Modify CAC Group Input Fields

8.10.4 Query CAC Group

This function is used to query Call Admission Control (CAC) Group information. Returned information is for either a single CAC Group or a set of CAC Groups depending on the input filter parameters.

Note: If no filters are specified, all CAC Group information is returned.

Table 168 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
	N/A	subRC_SUCCESS subRC_APP_CAC_INVALID_BG_NAME subRC_APP_CAC_INVALID_BO_NAME

Table 168 Query CAC Group Return Codes

Table 169 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
BusinessGroupName	The name of the Business Group associated with the Call Admission Control (CAC) Group to be queried. This is an optional filter for the query. If a BusinessGroupName is specified, information is returned for only that particular business group.	0..30	Character String	“”
BranchOfficeName	The name of the Branch Office associated with the Call Admission Control (CAC) Group to be queried. This is an optional filter for the query. If a BranchOfficeName is specified, information is returned for only that particular branch office.	0..30	Character String	“”
InterfaceWSDL	Refer to Section 5.1.1, “The Interface WSDL and Interface Release Control”, on page 61.			
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			

Table 169 Query CAC Group Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
CACGroupName	The name of the CAC Group to be queried. This is an optional filter for the query. If a CACGroupName is specified, information is returned for only that particular CAC Group. CAC Group Name filter also supports wildcards: • * for 0 or more matches, and, • ? exact one match. In this case, one or more CAC Groups could be returned. This input is logical ANDed with all the other filter inputs (if any). A CACGroupName with wildcards cannot co-exist in the same filter with a CACGroupMember filter (see below description of CACGroup Member).	Please refer to CACGroupName in Section 8.10.1, "Create CAC Group" for these values.		N/A
CACGroupType	Indicates the CAC Group Type to be queried. This is an optional filter for the query. If a CAC Group Type is specified, information is returned only for CAC Groups of the specified type.	enum	SubNetType, DNAddrType, IpAddrType	N/A
RouterIpAddress	An IP Address string representing the router associated with the CAC Group serving the WAN. Providing this parameter allows for the query of a CAC Group based on a known Router IP Address.	1..30	Character string	""
CACGroupMember	The name of a CAC Group Member. With this filter, the info of the CAC Group, that this member is belonging to, can be retrieved. A specific member can only belong to one CAC Group.	1...31	Character String	N/A
BGName	Business Group Name to be associated with CAC Group	Char	0…30	optional
BOName	Branch Office Name to be associated with CAC Group	char	0…30	Optional

Table 169 Query CAC Group Input Fields (Sheet 2 of 2)

Table 170 lists the output fields for this method.

Field Name	Description	Length	Range/Values	Default Value
An array of CACGroupInfo elements, each consisting of the following attributes:				
BusinessGroupName	The name of the Business Group associated with the Call Admission Control (CAC) Group	Please refer to the definition of BusinessGroupName in Section 8.10.1, “Create CAC Group”.		N/A
BranchOfficeName	The name of the Branch Office associated with the Call Admission Control (CAC) Group	Please refer to the definition of BusinessGroupName in Section 8.10.1, “Create CAC Group”		N/A
CACGroupName	The name of the CAC Group.	Please refer to the definition of CACGroupName in Section 8.10.1, “Create CAC Group”.		N/A
CACGroupType	Indicates the CAC Group Type to be queried. This is an optional filter for the query. If a CAC Group Type is specified, information is returned only for CAC Groups of the specified type..	Enum	CACGroupSubNetType, CACGroupDNAddrType, CACGroupIpAddrType CACGroupParentCACGroups CACGroupLocationDomainType (V5.00.18.01)	N/A
AddressArray	One or more address strings. The valid values for these strings are based on CACGroupType	Please refer to the definition of AddressArray in Section 8.10.1, “Create CAC Group”.		N/A
DynamicCACRouterInfoData.DynamicCACRouterAddress.RouterIpAddress	An IP Address string representing the router associated with the CAC Group serving the WAN.	Please refer to the definition of RouterIpAddress in Section 8.10.1, “Create CAC Group”.		N/A
DynamicCACRouterInfoData.DynamicCACRouterAddress.InterfaceName	This identifies the interface within the router. The string is of the form “eth0”, “eth1” etc.	Please refer to the definition of InterfaceName in Section 8.10.1, “Create CAC Group”.		N/A
DynamicCACRouterInfoData.CACLinkStatusTimeStamp	The time when the status for the link was changed to “LinkDown”. If the CACGroup has been created and no status notification update is provided, then this time will default to an empty string. When the link status is changed to “LinkUp”, this field will be cleared.	time stamp		N/A
DynamicCACRouterInfoData.CACPrimaryLinkStatus	Indicates the status of the link.	enum	LinkUp LinkDown	N/A
parentCACGroupName	The name of the parent CAC Group Name	1...31		N/A

Table 170 Query CAC Group Output Fields

8.10.5 Add CAC Group Address

This method adds an address to a Call Admission Control (CAC) Group. Table 171 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
TBD	N/A	subRC_SUCCESS

Table 171 Add CAC Group Address Return Codes

Table 172 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
CACGroupName	The name of the CAC Group to be modified.	Please refer to the definition of CACGroupName in Section 8.10.1, “Create CAC Group”.		required
AddressArray	One or more new address strings to be added to the CAC Group. Note that the addresses specified in the AddressArray should conform to the existing CACGroupType of the CAC Group to be modified. For example, to add new addresses to a CAC Group with CACGroupType=IpAddrType, all the new addresses specified in AddressArray must be valid IP addresses.	Please refer to the definition of AddressArray in Table 161 on page 181 above.		N/A

Table 172 Add CAC Group Address Input Fields

8.10.6 Remove CAC Group Addr

This method removes an address from a Call Admission Control (CAC) Group. Table 173 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
TBD	N/A	subRC_SUCCESS

Table 173 Remove CAC Group Addr Return Codes

Table 174 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, "Field Definitions", on page 67.		
CACGroupName	The name of the CAC Group to be modified.	Please refer to the definition of CACGroupName in Section 8.10.1, "Create CAC Group".		required
AddressArray	One or more new address strings to be removed from the CAC Group.	Please refer to the definition of AddressArray in Table 161 on page 181 above.		N/A

Table 174 Remove CAC Group Addr Input Fields

8.10.7 Get Ancestor Tree For CAC Group

For a given CAC Group, this function returns a list of ancestors (parent, grandparent, etc.).

Table 175 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
12.00.01.29	N/A	subRC_SUCCESS subRC_APP_CAC_INVALID_CAC_GROUP_NAME subRC_APP_CAC_GROUP_QUERY_GET_ANCESTOR_NDM_ERROR subRC_APP_CAC_NON_EXISTENT_CAC_GROUP subRC_INT_CAC_GROUP_DB_INCONSISTENCY subRC_SYS_CAC_ALLOC_ERR

Table 175 Get Ancestor Tree for CAC Group Return Codes

Table 176 lists the input fields for this method. The output of this SOAP request is an array of strings, each representing the name of a CAC Group.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.		Please refer to the definition of Operator ID in Section 5.4.2, "Field Definitions", on page 67.	
InterfaceWSDL	Refer to Section 5.1.1, "The Interface WSDL and Interface Release Control", on page 61.			
CACGroupName	The name of the CAC Group to be queried. This is an optional filter for the query. If a CACGroupName is specified, information is returned for only that particular CAC Group.		Please refer to the definition of CACGroupName in Section 8.10.1, "Create CAC Group".	**N/A**

Table 176 Get Ancestor Tree for CAC Group Input Fields

8.11 CAC Policy

8.11.1 Create CAC Policy

This method creates a Call Admission Control (CAC) Policy. Table 177 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
TBD	N/A	subRC_SUCCESS
V4.00.01.04		31700050 subRC_APP_CAC_POLICY_CREATE_ATTRIBUTES_ERROR Error while creating attributes for a CAC policy.

Table 177 Create CAC Policy Return Codes

Table 178 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
CACGroupName	The name of the CAC Group for which this policy is to be created.	Please refer to the definition of CACGroupName in Section 8.10.1, “Create CAC Group”.		required
CACPolicyType	Indicates the CAC policy type. Policy type is a bitmap, where its bits represent the FaxPolicy, VoicePolicy, and VideoPolicy.	integer	0x0004 (VideoPolicyType) 0x0005 (Voice & VideoPolicyType) 0x0006 (FaxPolicyType) 0x0007 (all three of the above)	required
CACPolicyLimitType	Indicates the type of limit specified by the CAC policy.	enum	CACBandwidthLimitType CACNumCallsLimitType CACNumCallsAndBWLimitType	required
CallLimit	If the CAC policy is to impose a limit on the number of calls, this parameter should be specified.	integer	0..2147483647 Note this parameter is required if CACPolicyLimitType is CACNumCallsLimitType or CACNumCallsAndBWLimitType	-1
BandWidthLimit	If the CAC policy is to impose a limit on the bandwidth, this parameter should be specified.	integer	0..2147483647 Note this parameter is required if CACPolicyLimitType is CACBandwidthLimitType or CACNumCallsAndBWLimitType	-1

Table 178 Create CAC Policy Input Fields (Sheet 1 of 4)

Field Name	Description	Length	Range/Values	Default Value
CACPolicyOptionFlag	CACMediaServerAllow change to CACMediaServerIgnore This field is deprecated from WSDL version V4.00.01.04 and above. The field CACPolicyAttributes (please see below) should be used instead.	enum	Change value: CACMediaServerAllow to CACMediaServerIgnore	CACNoOption
CACPolicyAttributes	This field replaces the CACPolicyOptionFlag field. It is used to provision the available options to a policy. The existing options: 1) Ignore calls to media server for announcements and tones 2) Allow answered calls when there is not enough bandwidth 3) Allow video call to proceed as an audio only call when not enough bandwidth	array of Boolean flags	CACPolicyIgnoreMediaServer CACPolicyAllowAnsweredCalls CACPolicyAllowVideoCallsAsAudioOnly	all false
DynamicCACPolicyInfo.SecondaryCallLimit	Specifies the secondary Call Limit capacity. This parameter has the same validations as the CallLimit parameter with respect to the CACPolicyLimitType. It can be entered only if the CACGroup has a RouterIPAddress configured. This parameter is mandatory if the RouterIPAddress in the CACGroup has been configured and the CACPolicyType is “CallLimit” or “Both”.	integer		-1

Table 178 Create CAC Policy Input Fields (Sheet 2 of 4)

Field Name	Description	Length	Range/Values	Default Value
DynamicCACPolicyInfo. SecondaryBandWidthLimit	Specifies the secondary Bandwidth Limit capacity. This parameter has the same validations as the BandWidthLimit parameter with respect to the CACPolicyLimitType. It can be entered only if the CACGroup has a RouterIPAddress configured. This parameter is mandatory if the RouterIPAddress in the CACGroup has been configured and the CACPolicyType is “BandwidthLimit” or “Both”.	integer		-1
CACPolicyCodecsAllowed	A list of CODEC flags, each one set to true or false to indicate whether or not the CODEC is allowed for this CAC policy. This information is used to limit the CODECs allowed for voice calls over the bandwidth-limited link to optimize the usage of the bandwidth and allow more simultaneous connections while still guaranteeing an acceptable quality of service.	array of Boolean flags	CACPolicyAllCodec CACPolicyCodecG711ALaw CACPolicyCodecG711ULaw CACPolicyCodecG722 CACPolicyCodecG7231 CACPolicyCodecG726_16 CACPolicyCodecG726_24 CACPolicyCodecG726_32 CACPolicyCodecG726_40 CACPolicyCodecG728 CACPolicyCodecG729 CACPolicyCodecILBC CACPolicyCodecAMR CACPolicyCodecAMR_WB CACPolicyCodecAAC_LC CACPolicyCodecG7221_24 CACPolicyCodecG7221_32 CACPolicyCodecG7221_48	CACPolicy AllCodec = true
IgnoreMediaServer	This value is for ignoring the Media Server.	Boolean	True/False	
AllowAnsweredCalls	This value is for allowing answered calls.	Boolean	True/False	
AllowVideoCallsAsAudioOnly	This value is for allowing Video Calls As Audio Only.	Boolean	True/False	
GenerateAlarm	Enable or disable generation of alarms when the CACPolicyLimitType exceeds its thresholds. Limit type policies are: 1). bandwidth limit and/or 2). number of calls Optional.	Boolean	True/False	False

Table 178 Create CAC Policy Input Fields (Sheet 3 of 4)

Field Name	Description	Length	Range/Values	Default Value
AlarmLowThreshold	Sets the low percentage threshold for the selected policy type. When exceeded an alarm will be generated if GenerateAlarm is set to true. Optional.	Integer	0-90	60*
AlarmHighThreshold	Sets the high percentage threshold for the selected policy type. When exceeded an alarm will be generated if GenerateAlarm is set to true. Optional.	Integer	10-100	80*
CACPolicyVideoCodecsAllowed	A list of VIDEO CODEC flags, eachone set to true or false toindicate whether or not theVIDEO CODEC is allowed for thisCAC policy.This information is used tolimit the VIDEO CODECs allowed forcalls over thebandwidth-limited link tooptimize the usage of thebandwidth and allow moresimultaneous connectionswhile still guaranteeing anacceptable quality of service.	array ofBooleanflags	CACPolicyAllCodec; CACPolicyCodecH263; CACPolicyCodecH264; CACPolicyCodecVP8;	N/A
MaximumProfileLevel	A maximum allowed profile level for H.264 shall be considered and this will be effective regardless of any codes restriction configuration.It is a series of combination of bitrates for H264	enum	Level_Unknown=-1,Level_1, Level_1_b, Level_1_1, Level_1_2, Level_1_3, Level_2, Level_2_1, Level_2_2, Level_3, Level_3_1, Level_3_2, Level_4, Level_4_1, Level_4_2, Level_5, Level_5_1	Level_2_1
MaxVideoCodecBandWidth	Maximum bandwidth allowed for video codecs.	integer	Allowed range for Video Codec bandwidth is [64000 to 240000000] (bps)	4000 000 (bps)

Table 178 Create CAC Policy Input Fields (Sheet 4 of 4)

* In reality SOAP displays “0” for the defaults when they are in use. These defaults are 60/80.

8.11.2 Delete CAC Policy

This method deletes a Call Admission Control (CAC) Policy. Table 179 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
TBD	N/A	subRC_SUCCESS

Table 179 Delete CAC Policy Return Codes

Table 180 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
CACGroupName	The name of the CAC Group from which this policy to be deleted.	Please refer to the definition of CACGroupName in Section 8.10.1, “Create CAC Group”.		required
CACPolicyType	Indicates the CAC policy type. Policy type is a bitmap, where its bits represent the FaxPolicy, VoicePolicy, and VideoPolicy.	integer	0x0004 (VideoPolicyType) 0x0005 (Voice & VideoPolicyType) 0x0006 (FaxPolicyType) 0x0007 (all three of the above)	required

Table 180 Delete CAC Policy Input Fields

8.11.3 Modify CAC Policy

This method modifies a Call Admission Control (CAC) Policy. Table 181 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
TBD	N/A	subRC_SUCCESS
V4.00.01.04		31700051 subRC_APP_CAC_POLICY_MODIFY_ATTRIBUTES_ERROR Error while modifying attributes for a CAC policy.

Table 181 Modify CAC Policy Return Codes

Table 182 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
CACGroupName	The name of the CAC Group whose policy is to be modified.	Please refer to the definition of CACGroupName in Section 8.10.1, “Create CAC Group”.		required

Table 182 Modify CAC Policy Input Fields (Sheet 1 of 4)

Field Name	Description	Length	Range/Values	Default Value
CACPolicyType	Indicates the CAC policy type. Policy type is a bitmap, where its bits represent the FaxPolicy, VoicePolicy, and VideoPolicy.	integer	0x0004 (VideoPolicyType) 0x0005 (Voice & VideoPolicyType) 0x0006 (FaxPolicyType) 0x0007 (all three of the above)	required
CACPolicyLimitType	Indicates the type of limit specified by the CACPolicy.	enum	CACBandwidthLimitType CACNumCallsLimitType CACNumCallsAndBWLimitType	N/A
CallLimit	If the CAC policy is to impose a limit on the number of calls, this parameter should be specified.	integer	0..2147483647	N/A
BandWidthLimit	If the CAC policy is to impose a limit on the bandwidth, this parameter should be specified.	integer	0..2147483647	N/A
CACPolicyOptionFlag	CACMediaServerAllow change to CACMediaServerIgnore This field is deprecated from WSDL version V4.00.01.04 and above. The field CACPolicyAttributes (please see below) should be used instead.	enum	Change value: CACMediaServerAllow to CACMediaServerIgnore	CACNoOption
CACPolicyAttributes	This field replaces the CACPolicyOptionFlag field. It is used to provision the available options to a policy. The existing options: 1) Ignore calls to media server for announcements and tones 2) Allow answered calls when there is not enough bandwidth 3) Allow video call to proceed as an audio only call when not enough bandwidth	array of Boolean flags	CACPolicyIgnoreMediaServer CACPolicyAllowAnsweredCalls CACPolicyAllowVideoCallsAsAudioOnly	all false
DynamicCACPolicyInfo.SecondaryCallLimit	Specifies the secondary Call Limit capacity. This parameter has the same validations as the CallLimit parameter with respect to the CACPolicyLimitType. It can be entered only if the CACGroup has a RouterIPAddress configured.	integer		N/A

Table 182 Modify CAC Policy Input Fields (Sheet 2 of 4)

Field Name	Description	Length	Range/Values	Default Value
DynamicCACPolicyInfo. SecondaryBandWidthLimit	Specifies the secondary Bandwidth Limit capacity. This parameter has the same validations as the BandWidthLimit parameter with respect to the CACPolicyLimitType. It can be entered only if the CACGroup has a RouterIPAddress configured.	integer		N/A
IgnoreMediaServer		Boolean	True/False	
AllowAnsweredCalls		Boolean	True/False	
AllowVideoCallsAsAudioOnly		Boolean	True/False	
GenerateAlarm	Enable or disable generation of alarms when the CACPolicyLimitType exceeds its thresholds. Limit type policies are: 1). bandwidth limit and/or 2). number of calls Optional.	Boolean	True/False	False
AlarmLowThreshold	Sets the low percentage threshold for the selected policy type. When exceeded an alarm will be generated if GenerateAlarm is set to true. Optional.	Integer	0-90	60*
AlarmHighThreshold	Sets the high percentage threshold for the selected policy type. When exceeded an alarm will be generated if GenerateAlarm is set to true. Optional.	Integer	10-100	80*
CACPolicyVideoCodecsAllowed	A list of VIDEO CODEC flags, eachone set to true or false toindicate whether or not theVIDEO CODEC is allowed for thisCAC policy.This information is used tolimit the VIDEO CODECs allowed forcalls over thebandwidth-limited link tooptimize the usage of thebandwidth and allow moresimultaneous connectionswhile still guaranteeing anacceptable quality of service.	array ofBooleanflags	CACPolicyAllCodec; CACPolicyCodecH263; CACPolicyCodecH264; CACPolicyCodecVP8;	N/A

Table 182 Modify CAC Policy Input Fields (Sheet 3 of 4)

Field Name	Description	Length	Range/Values	Default Value
MaximumProfileLevel	A maximum allowed profile level for H.264 shall be considered and this will be effective regardless of any codes restriction configuration.It is a series of combination of bitrates for H264	enum	Level_Unknown=-1,Level_1, Level_1_b, Level_1_1, Level_1_2, Level_1_3, Level_2, Level_2_1, Level_2_2, Level_3, Level_3_1, Level_3_2, Level_4, Level_4_1, Level_4_2, Level_5, Level_5_1	Level_2_1
CACPolicyCodecsAllowed	A list of CODEC flags, eachone set to true or false toindicate whether or not theCODEC is allowed for thisCAC policy.This information is used tolimit the CODECs allowed forvoice calls over thebandwidth-limited link tooptimize the usage of thebandwidth and allow moresimultaneous connectionswhile still guaranteeing anacceptable quality of service.	array ofBooleanflags	CACPolicyAllCodec CACPolicyCodecG711ALaw CACPolicyCodecG711ULaw CACPolicyCodecG722 CACPolicyCodecG7231 CACPolicyCodecG726_16 CACPolicyCodecG726_24 CACPolicyCodecG726_32 CACPolicyCodecG726_40 CACPolicyCodecG728 CACPolicyCodecG729 CACPolicyCodecILBC CACPolicyCodecAMR CACPolicyCodecAMR_WB CACPolicyCodecAAC_LC CACPolicyCodecG7221_24 CACPolicyCodecG7221_32 CACPolicyCodecG7221_48	CACPolicyAllCodec =true
MaxVideoCodecBandWidth	Maximum bandwidth allowed for video codecs.	integer	Allowed range for Video Codec bandwidth is [64000 to 240000000] (bps)	4000 000 (bps)

Table 182 Modify CAC Policy Input Fields (Sheet 4 of 4)

* In reality SOAP displays “0” for the defaults when they are in use. These defaults are 60/80.

Note: The absolute difference between low and high threshold must be equal to or greater than 10%.

8.11.4 Query CAC Policy

This function is used to query Call Admission Control (CAC) Policy information. Returned information is for either a single CAC Policy or a set of CAC Policies depending on the input filter parameters.

Note that if neither CACGroupName nor CACPolicyType is specified, all CAC Policy information is returned. Table 183 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
TBD	N/A	subRC_SUCCESS

Table 183 Query CAC Policy Return Codes

Table 184 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
InterfaceWSDL	Refer to Section 5.1.1, “The Interface WSDL and Interface Release Control”, on page 61.			
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			
CACGroupName	The name of the CAC Group to be queried. This is an optional filter for the query. If a CACGroupName is specified, information is returned for only that particular CAC Group.	Please refer to the definition of CACGroupName in Section 8.10.1, “Create CAC Group”.		
CACPolicyType	Indicates the CAC policy type. Policy type is a bitmap, where its bits represent the FaxPolicy, VoicePolicy, and VideoPolicy.	integer	0x0004 (VideoPolicyType) 0x0005 (Voice & VideoPolicyType) 0x0006 (FaxPolicyType) 0x0007 (all three of the above)	required
IgnoreMediaServer		Boolean	True/False	
AllowAnsweredCalls		Boolean	True/False	
AllowVideoCallsAsAudioOnly		Boolean	True/False	

Table 184 Query CAC Policy Input Fields

Table 185 lists the output fields for this method.

Field Name	Description	Length	Range/Values	Default Value
CACPolicyInfoArray	An array of structures, each of which contains all attributes of a particular CAC policy. Please refer to the list of attributes in Section 8.11.1, “Create CAC Policy” for details.			

Table 185 Query CAC Policy Output Fields

8.11.5 Get Bandwidth Usage for CAC Policy

This function is used to get the current bandwidth usage and number of concurrent calls for a Call Admission Control (CAC) Policy. Table 186 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
TBD	N/A	subRC_SUCCESS

Table 186 Get Bandwidth Usage for CAC Policy Return Codes

Table 187 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
InterfaceWSDL	Refer to Section 5.1.1, “The Interface WSDL and Interface Release Control”, on page 61.			
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			
CACGroupName	The name of the CAC Group to be queried. This is an optional filter for the query. If a CACGroupName is specified, information is returned for only that particular CAC Group.	Please refer to the definition of CACGroupName in Section 8.10.1, “Create CAC Group”.		required
CACPolicyType	Indicates the CAC Policy Type. Policy Type is a bitmap, where the last two bits represent the FaxPolicy and VoicePolicy respectively. In the future this bitmap may be extended to support additional types.	integer	0x0001 (VoicePolicyType) 0x0002 (FaxPolicyType) 0x0003 (both of the above)	required

Table 187 Get Bandwidth Usage for CAC Policy Input Fields

Table 188 lists the output fields for this method.

Field Name	Description	Length	Range/Values	Default Value
BandWidthUse	Actual bandwidth used for this CAC policy.			N/A
NumConCurrentCalls	The number of current concurrent calls for this CAC policy.			N/A

Table 188 Get Bandwidth Usage for CAC Policy Output Fields

8.11.6 Create Group To Group CAC Policy

This method creates a Group to Group Call Admission Control (CAC) Policy. This SOAP request is very similar to the Create CAC Policy request, except that in this case two CAC Group Names are specified for which the GToGCACPolicy is to be defined.

Table 189 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
12.00.01.29	N/A	subRC_SUCCESS subRC_APP_CAC_INVALID_CAC_GROUP_NAME subRC_APP_CAC_GROUP_NAMES_SAME_FOR_GROUPTOGROUP_CACPOLICY subRC_APP_CAC_INVALID_CAC_POLICY_TYPE subRC_APP_CAC_INVALID_CAC_POLICY_LIMIT_TYPE subRC_APP_CAC_BANDWIDTH_LIMIT_NOT_SPECIFED subRC_APP_CAC_INVALID_CAC_POLICY_BAND_LIMIT subRC_APP_CAC_CALL_LIMIT_NOT_SPECIFED subRC_APP_CAC_INVALID_CAC_POLICY_NUM_CALL_LIMIT subRC_APP_CAC_INVALID_CAC_POLICY_OPTION_FLAG subRC_APP_GROUPTOGROUP_CAC_POLICY_CREATE_ERROR
V4.00.01.04		31700052 subRC_APP_GTOC_POLICY_CREATE_ATTRIBUTES_ERROR Error while creating attributes for a group-to-group CAC policy.

Table 189 Create Group to Group CAC Policy Return Codes

Table 190 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
CACGroup1Name	The name of the first CAC Group for which this policy is to be created.	Please refer to the definition of CACGroupName in Section 8.10.1, “Create CAC Group”.		required
CACGroup2Name	The name of the second CAC Group for which this policy is to be created.	Please refer to the definition of CACGroupName in Section 8.10.1, “Create CAC Group”.		required
CACPolicyType	Indicates the CAC policy type. Policy type is a bitmap, where its bits represent the FaxPolicy, VoicePolicy, and VideoPolicy.	integer	0x0004 (VideoPolicyType) 0x0005 (Voice & VideoPolicyType) 0x0006 (FaxPolicyType) 0x0007 (all three of the above)	required
CACPolicyLimitType	Indicates the type of limit specified by the CACPolicy.	enum	CACBandwidthLimitType CACNumCallsLimitType CACNumCallsAndBWLimitType	required

Table 190 Create Group to Group CAC Policy Input Fields (Sheet 1 of 3)

Field Name	Description	Length	Range/Values	Default Value
CallLimit	If the CAC policy is to impose a limit on the number of calls, this parameter should be specified.	integer	0..2147483647 Note this parameter is required if CACPolicyLimitType is CACNumCallsLimitType or CACNumCallsAndBWLimitType	-1
BandWidthLimit	If the CAC policy is to impose a limit on the bandwidth, this parameter should be specified.	integer	0..2147483647 Note this parameter is required if CACPolicyLimitType is CACBandwidthLimitType or CACNumCallsAndBWLimitType	-1
CACPolicyOptionFlag	CACMediaServerAllow change to CACMediaServerIgnore This field is deprecated from WSDL version V4.00.01.04 and above. The field CACPolicyAttributes (please see below) should be used instead.	enum	Change value: CACMediaServerAllow to CACMediaServerIgnore	CACNoOption
CACPolicyAttributes	This field replaces the CACPolicyOptionFlag field. It is used to provision the available options to a policy. The existing options: 1) Ignore calls to media server for announcements and tones 2) Allow answered calls when there is not enough bandwidth 3) Allow video call to proceed as an audio only call when not enough bandwidth	array of Boolean flags	CACPolicyIgnoreMediaServer CACPolicyAllowAnsweredCalls CACPolicyAllowVideoCallsAsAudioOnly	all false

Table 190 Create Group to Group CAC Policy Input Fields (Sheet 2 of 3)

Field Name	Description	Length	Range/Values	Default Value
CACPolicyCodecsAllowed	A list of CODEC flags, each one set to true or false to indicate whether or not the CODEC is allowed for this CAC policy. This information is used to limit the CODECs allowed for voice calls over the bandwidth-limited link to optimize the usage of the bandwidth and allow more simultaneous connections while still guaranteeing an acceptable quality of service.	array of Boolean flags	CACPolicyAllCodec CACPolicyCodecG711ALaw CACPolicyCodecG711ULaw CACPolicyCodecG722 CACPolicyCodecG7231 CACPolicyCodecG726_16 CACPolicyCodecG726_24 CACPolicyCodecG726_32 CACPolicyCodecG726_40 CACPolicyCodecG728 CACPolicyCodecG729 CACPolicyCodecILBC CACPolicyCodecAMR CACPolicyCodecAMR_WB CACPolicyCodecAAC_LC CACPolicyCodecG7221_24 CACPolicyCodecG7221_32 CACPolicyCodecG7221_48	CACPolicyAllCodec = true
IgnoreMediaServer		Boolean	True/False	
AllowAnsweredCalls		Boolean	True/False	
AllowVideoCallsAsAudioOnly		Boolean	True/False	
CACPolicyVideoCodecsAllowed	A list of VIDEO CODEC flags, eachone set to true or false toindicate whether or not theVIDEO CODEC is allowed for thisCAC policy.This information is used tolimit the VIDEO CODECs allowed forcalls over thebandwidth-limited link tooptimize the usage of thebandwidth and allow moresimultaneous connectionswhile still guaranteeing anacceptable quality of service.	array ofBooleanflags	CACPolicyAllCodec; CACPolicyCodecH263; CACPolicyCodecH264; CACPolicyCodecVP8;	N/A
MaximumProfileLevel	A maximum allowed profile level for H.264 shall be considered and this will be effective regardless of any codes restriction configuration.It is a series of combination of bitrates for H264	enum	Level_Unknown=-1, Level_1, Level_1_b, Level_1_1, Level_1_2, Level_1_3, Level_2, Level_2_1, Level_2_2, Level_3, Level_3_1, Level_3_2, Level_4, Level_4_1, Level_4_2, Level_5, Level_5_1	Level_2_1

Table 190 Create Group to Group CAC Policy Input Fields (Sheet 3 of 3)

8.11.7 Delete Group To Group CAC Policy

This method deletes a Group To Group Call Admission Control (CAC) Policy. This SOAP request is very similar to the Delete CAC Policy request, except that in this case two CACGroup Names are required to identify the GToGCACPolicy.

Table 191 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
12.00.01.29	N/A	subRC_SUCCESS subRC_APP_CAC_INVALID_CAC_GROUP_NAME subRC_APP_CAC_INVALID_CAC_POLICY_TYPE subRC_APP_CAC_POLICY_DELETE_NDM_ERROR

Table 191 Delete Group to Group CAC Policy Return Codes

Table 192 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
CACGroup1Name	The name of the first CAC Group for this policy.	Please refer to the definition of CACGroupName in Section 8.10.1, “Create CAC Group”.		required
CACGroup2Name	The name of the second CAC Group for this policy.	Please refer to the definition of CACGroupName in Section 8.10.1, “Create CAC Group”.		required
CACPolicyType	Indicates the CAC policy type. Policy type is a bitmap, where its bits represent the FaxPolicy, VoicePolicy, and VideoPolicy.	integer	0x0004 (VideoPolicyType) 0x0005 (Voice & VideoPolicyType) 0x0006 (FaxPolicyType) 0x0007 (all three of the above)	required

Table 192 Delete Group to Group CAC Policy Input Fields

8.11.8 Modify Group To Group CAC Policy

This method modifies a Group To Group Call Admission Control (CAC) Policy. This SOAP request is very similar to the Modify CAC Policy request, except that in this case two CACGroup Names are required to identify the GToGCACPolicy.

Table 193 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
12.00.01.29	N/A	subRC_SUCCESS subRC_APP_CAC_INVALID_CAC_GROUP_NAME subRC_APP_CAC_INVALID_CAC_POLICY_TYPE subRC_APP_CAC_INVALID_CAC_POLICY_NUM_CALL_LIMIT subRC_APP_CAC_INVALID_CAC_POLICY_BAND_LIMIT subRC_APP_CAC_INVALID_CAC_POLICY_OPTION_FLAG subRC_APP_CAC_POLICY_MODIFY_NDM_ERROR
V4.00.01.04		31700052 subRC_APP_GTOC_POLICY_MODIFY_ATTRIBUTES_ERROR Error while modifying attributes for a group-to-group CAC policy.
V4.01.04.01		subRC_APP_CAC_INVALID_MAX_PROFILE_LEVEL

Table 193 Modify Group to Group CAC Policy Return Codes

Table 194 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
CACGroup1Name	The name of the first CAC Group for this policy.	Please refer to the definition of CACGroupName in Section 8.10.1, “Create CAC Group”.		required
CACGroup2Name	The name of the second CAC Group for this policy.	Please refer to the definition of CACGroupName in Section 8.10.1, “Create CAC Group”.		required
CACPolicyType	Indicates the CAC Policy Type. Policy Type is a bitmap, where the last two bits represent the FaxPolicy and VoicePolicy respectively. In the future this bitmap may be extended to support additional types.	integer	0x0001 (VoicePolicyType) 0x0002 (FaxPolicyType) 0x0003 (both of the above)	required
CACPolicyLimitType	Indicates the type of limit specified by the CACPolicy.	enum	CACBandwidthLimitType CACNumCallsLimitType CACNumCallsAndBWLimitType	N/A
CallLimit	If the CAC policy is to impose a limit on the number of calls, this parameter should be specified.	integer	0..2147483647	N/A
BandWidthLimit	If the CAC policy is to impose a limit on the bandwidth, this parameter should be specified.	integer	0..2147483647	N/A

Table 194 Modify Group to Group CAC Policy Input Fields (Sheet 1 of 3)

Field Name	Description	Length	Range/Values	Default Value
CACPolicyOptionFlag	Specifies options for the CAC policy. The IgnoreMediaServerCalls option implies that for this policy the Media Server calls should not be counted against the used bandwidth and number of calls for the policy. The NumCallsLimitType option allows the number of answered calls to exceed the limit. The third option indicates both of the above options.	enum	CACNoOption CACIgnoreMediaServerCalls CACNumCallsLimitType CACMediaServerIgnoreNumCallsExcedAllow	CACNoOption
CACPolicyCodecsAllowed	A list of CODEC flags, each one set to true or false to indicate whether or not the CODEC is allowed for this CAC policy. This information is used to limit the CODECs allowed for voice calls over the bandwidth-limited link to optimize the usage of the bandwidth and allow more simultaneous connections while still guaranteeing an acceptable quality of service.	array of Boolean flags	CACPolicyAllCodec CACPolicyCodecG711ALaw CACPolicyCodecG711ULaw CACPolicyCodecG722 CACPolicyCodecG7231 CACPolicyCodecG726_16 CACPolicyCodecG726_24 CACPolicyCodecG726_32 CACPolicyCodecG726_40 CACPolicyCodecG728 CACPolicyCodecG729 CACPolicyCodecILBC CACPolicyCodecAMR CACPolicyCodecAMR_WB CACPolicyCodecAAC_LC CACPolicyCodecG7221_24 CACPolicyCodecG7221_32 CACPolicyCodecG7221_48	CACPolicyAllCodec = true
IgnoreMediaServer		Boolean	True/False	
AllowAnsweredCalls		Boolean	True/False	
AllowVideoCallsAsAudioOnly		Boolean	True/False	
CACPolicyVideoCodecsAllowed	A list of VIDEO CODEC flags, eachone set to true or false toindicate whether or not theVIDEO CODEC is allowed for thisCAC policy.This information is used tolimit the VIDEO CODECs allowed forcalls over thebandwidth-limited link tooptimize the usage of thebandwidth and allow moresimultaneous connectionswhile still guaranteeing anacceptable quality of service.	array ofBooleanflags	CACPolicyAllCodec; CACPolicyCodecH263; CACPolicyCodecH264; CACPolicyCodecVP8;	N/A

Table 194 Modify Group to Group CAC Policy Input Fields (Sheet 2 of 3)

Field Name	Description	Length	Range/Values	Default Value
MaximumProfileLevel	A maximum allowed profile level for H.264 shall be considered and this will be effective regardless of any codes restriction configuration.It is a series of combination of bitrates for H264	enum	Level_Unknown=-1, Level_1, Level_1_b, Level_1_1, Level_1_2, Level_1_3, Level_2, Level_2_1, Level_2_2, Level_3, Level_3_1, Level_3_2, Level_4, Level_4_1, Level_4_2, Level_5, Level_5_1	Level_2_1

Table 194 Modify Group to Group CAC Policy Input Fields (Sheet 3 of 3)

8.11.9 Query Group To Group CAC Policy

This function is used to query Group To Group Call Admission Control (CAC) Policy information. Returned information is for either a single CAC Policy or a set of CAC Policies depending on the input filter parameters. This SOAP request is very similar to the Query CAC Policy request, except that in this case two CACGroup Names may be provided as filter criteria for the query.

Note that if neither CACGroup1Name, CACGroup2Name, nor CACPolicyType is specified, all Group To Group CAC Policy information is returned.

Table 195 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
12.00.01.29	N/A	subRC_SUCCESS subRC_APP_CAC_INVALID_CAC_GROUP_NAME subRC_APP_CAC_INVALID_CAC_POLICY_TYPE subRC_APP_CAC_POLICY_QUERY_NDM_ERROR subRC_SYS_CAC_ALLOC_ERR

Table 195 Query Group to Group CAC Policy Return Codes

Table 196 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
InterfaceWSDL	Refer to Section 5.1.1, “The Interface WSDL and Interface Release Control”, on page 61.			

Table 196 Query Group to Group CAC Policy Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			
CACGroup1Name	The name of the first CAC Group on which to filter for the query. If a CACGroup1Name is specified, information is returned for only policies having that particular CACGroup1Name.	Please refer to the definition of CACGroupName in Section 8.10.1, “Create CAC Group”.		“”
CACGroup2Name	The name of the second CAC Group on which to filter for the query. If a CACGroup2Name is specified, information is returned for only policies having that particular CACGroup2Name.	Please refer to the definition of CACGroupName in Section 8.10.1, “Create CAC Group”.		“”
CACPolicyType	Indicates the CAC policy type. Policy type is a bitmap, where its bits represent the FaxPolicy, VoicePolicy, and VideoPolicy.	integer	0x0004 (VideoPolicyType) 0x0005 (Voice & VideoPolicyType) 0x0006 (FaxPolicyType) 0x0007 (all three of the above)	required
IgnoreMediaServer		Boolean	True/False	
AllowAnsweredCalls		Boolean	True/False	
AllowVideoCallsAsAudioOnly		Boolean	True/False	

Table 196 Query Group to Group CAC Policy Input Fields (Sheet 2 of 2)

Table 197 lists the output fields for this method.

Field Name	Description	Length	Range/Values	Default Value
GToGCACPolicyInfoArray	An array of structures, each of which contains all attributes of a particular Group To Group CAC Policy. Please refer to the list of attributes in Section 8.11.6, “Create Group To Group CAC Policy” for details.			

Table 197 Query Group to Group CAC Policy Output Fields

8.11.10 Get Bandwidth Usage for Group to Group CAC Policy

This function is used to get the current bandwidth usage and number of concurrent calls for a Group To Group Call Admission Control (CAC) Policy. This SOAP request is very similar to the Get Bandwidth Usage For CAC Policy request, except that in this case two CACGroup Names are required to identify the GToGCACPolicy

Table 198 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
12.00.01.29	N/A	subRC_SUCCESS subRC_APP_CAC_INVALID_CAC_GROUP_NAME subRC_APP_CAC_INVALID_CAC_POLICY_TYPE subRC_APP_CAC_POLICY_QUERY_BW_USAGE_NDM_ERROR subRC_APP_CAC_NON_EXISTENT_GROUPTOGROUP_CAC_POLICY

Table 198 Get BW Usage for Group to Group CAC Policy Return Codes

Table 199 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
InterfaceWSDL	Refer to Section 5.1.1, “The Interface WSDL and Interface Release Control”, on page 61.			
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			
CACGroup1Name	The name of the first CAC Group for this policy whose bandwidth usage information is to be retrieved.	Please refer to the definition of CACGroupName in Section 8.10.1, “Create CAC Group”.		required
CACGroup2Name	The name of the second CAC Group for this policy.	Please refer to the definition of CACGroupName in Section 8.10.1, “Create CAC Group”.		required
CACPolicyType	Indicates the CAC Policy Type. Policy Type is a bitmap, where the last two bits represent the FaxPolicy and VoicePolicy respectively. In the future this bitmap may be extended to support additional types.	integer	0x0001 (VoicePolicyType) 0x0002 (FaxPolicyType) 0x0003 (both of the above)	required

Table 199 Get BW Usage for Group to Group CAC Policy Input Fields

Table 200 lists the output fields for this method.

Field Name	Description	Length	Range/Values	Default Value
BandWidthUse	Actual bandwidth used for this CAC policy.			N/A
NumConCurrentCalls	The number of current concurrent calls for this CAC policy.			N/A

Table 200 Get BW Usage for Group to Group CAC Policy Output Fields

8.11.11 Query CAC Monitoring

This function is used to get the current bandwidth usage and number of concurrent calls for a Group To Group Call Admission Control (CAC) Policy. This SOAP request is very similar to the Get Bandwidth Usage For CAC Policy request, except that in this case two CACGroup Names are required to identify the GToGCACPolicy

A Soap Server interface that shall return the following information for each configured CAC Policy:

- CAC Policy ID
- CAC Policy Type (Traffic Type)
- CAC Group 1 Name
- CAC Group 2 Name
- CAC Policy Limit Type
- Call Limit (The API returns the limit which in currently in effect, primary or secondary)
- Bandwidth Limit (The API returns the limit which in currently in effect, primary or secondary)
- Number of Concurrent Calls
- Bandwidth Used

The same API is utilized to retrieve both the regular CAC Policies and the Group-To-Group CAC Policies.

Table 201 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
V5.00.01.09	N/A	subRC_APP_SORM_OPERATOR_INVALID subRC_APP_CP_ERROR_GET_CAC_LIST subRC_APP_CAC_MONITORING_QUERY_ERROR subRC_APP_CAC_MONITORING_PAGING_OUT_OF_BOUNDS

Table 201 QueryCacMonitoring Return Codes

Table 202 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		required
InterfaceWSDL	Refer to Section 5.1.1, “The Interface WSDL and Interface Release Control”, on page 61.			
Page Number	number of the retrieved result set	Integer		
Page Size	Page size of the retrieved result set	Integer		

Table 202 QueryCacMonitoring Input Fields

8.12 Time Destination

8.12.1 Create Time Destination

This function is used to create a Time Destination.

The following relationships exist among Time Destinations and their related objects:

A Time Destination is created first, then one or more Day Schedules are created within a particular Time Destination. A Period Schedule is a period of time within a particular day, so a Day Schedule is made up of a set of Period Schedules. When a Day Schedule is full of Period Schedules such that the periods specified cover the entire day, the Day Schedule is assigned to a particular day of the week. Day Schedules can later be unassigned or modified if desired. If a Day Schedule is not assigned, it can be removed. To see the complete set of assignments of Day Schedules to the days of the week for a particular Time Destination, the QueryWeeklySchedule request is used. See the following subsections for detailed information about all of these operations.

Table 203 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 203 Create Time Destination Return Codes

Table 204 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Name	The name of the Time Destination to create.	1..15	Character String	required

Table 204 Create Time Destination Input Fields

8.12.2 Remove Time Destination

This function is used to create a Time Destination. Table 205 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 205 Remove Time Destination Return Codes

Table 206 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Name	The name of the Time Destination to remove.	1..15	Character String	required

Table 206 Remove Time Destination Input Fields

8.12.3 Query Time Destination

This function is used to retrieve and display a Time Destination. If the Name parameter is supplied, only that Time Destination is retrieved; otherwise the complete list of existing Time Destination is retrieved.

Table 207 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 207 Query Time Destination Return Codes

Table 208 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Specific	Indicates type of query. If True, the request is for a specific TimeDestination; if False, for all TimeDestinations.	Refer to the paragraph at the beginning of Chapter 8.		
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			
Name	The name of the Time Destination to retrieve.	1..15	Character String	N/A

Table 208 Query Time Destination Input Fields

8.13 Day Schedule

8.13.1 Create Day Schedule

This function is used to provision a specific Day Schedule within a Time Destination. The schedules assigned within the time destination should cover one whole week. Note that the name of the Day Schedule itself need not be unique in the system - the day schedule is uniquely identified by the combination of Day Schedule and Time Destination.

Note that this operation does not assign a Day Schedule to the Time Destination, it only creates it and uniquely identifies it within the Time Destination so that it can be assigned using the AssignDaySchedule operation.

Table 209 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 209 Create Day Schedule Return Codes

Table 210 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		

Table 210 Create Day Schedule Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
Name	The name of the Day Schedule to create. This uniquely identifies a day schedule within a time destination.	1..15	Character String	required
TimeDestinationName	The name of an existing Time Destination.	1..15	Character String	required

Table 210 Create Day Schedule Input Fields (Sheet 2 of 2)

8.13.2 Assign Day Schedule

This function is used to assign one or more Day Schedules to a Time Destination.

Note that any day schedule that is assigned must be fully allocated, meaning that it must contain period schedules that cover exactly 24 hours (1440 minutes).

Table 211 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 211 Assign Day Schedule Return Codes

Table 212 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
TimeDestinationName	The name of the Time Destination to which Day Schedules are to be assigned.	1..15	Character String	required
DayScheduleNameSun	The name of an existing Day Schedule to assign to the Time Destination for Sunday.	1..15	Character String	N/A
DayScheduleNameMon	The name of an existing Day Schedule to assign to the Time Destination for Monday.	1..15	Character String	N/A
DayScheduleNameTue	The name of an existing Day Schedule to assign to the Time Destination for Tuesday.	1..15	Character String	N/A
DayScheduleNameWed	The name of an existing Day Schedule to assign to the Time Destination for Wednesday.	1..15	Character String	N/A

Table 212 Assign Day Schedule Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
DayScheduleNameThu	The name of an existing Day Schedule to assign to the Time Destination for Thursday.	1..15	Character String	N/A
DayScheduleNameFri	The name of an existing Day Schedule to assign to the Time Destination for Friday.	1..15	Character String	N/A
DayScheduleNameSat	The name of an existing Day Schedule to assign to the Time Destination for Saturday.	1..15	Character String	N/A

Table 212 Assign Day Schedule Input Fields (Sheet 2 of 2)

8.13.3 Unassign Day Schedule

This function is used to unassign one or more Day Schedules from a Time Destination.

Note that this operation only unassigns the Day Schedule, but does not remove it—it will still be available within the Time Destination for future assignments.

Table 213 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 213 Unassign Day Schedule Return Codes

Table 214 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
TimeDestinationName	The name of the Time Destination from which Day Schedules are to be unassigned.	1..15	Character String	required
UnassignAll	If True, indicates that all Day Schedules are to be unassigned.	Boolean	True, False	False
UnassignSun	If True, indicates that the Day Schedule specified as DayScheduleNameSun is to be unassigned.	Boolean	True, False	False

Table 214 Unassign Day Schedule Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
UnassignMon	If True, indicates that the Day Schedule specified as DayScheduleNameMon is to be unassigned.	Boolean	True, False	False
UnassignTue	If True, indicates that the Day Schedule specified as DayScheduleNameTue is to be unassigned.	Boolean	True, False	False
UnassignWed	If True, indicates that the Day Schedule specified as DayScheduleNameWed is to be unassigned.	Boolean	True, False	False
UnassignThu	If True, indicates that the Day Schedule specified as DayScheduleNameThu is to be unassigned.	Boolean	True, False	False
UnassignFri	If True, indicates that the Day Schedule specified as DayScheduleNameFri is to be unassigned.	Boolean	True, False	False
UnassignSat	If True, indicates that the Day Schedule specified as DayScheduleNameSat is to be unassigned.	Boolean	True, False	False

Table 214 Unassign Day Schedule Input Fields (Sheet 2 of 2)

8.13.4 Modify Day Schedule

This function is used to modify one or more Day Schedules for a Time Destination. In other words, it changes the existing assignment for each specified day from one Day Schedule to another, accomplishing the same functionality that could be done by using Unassign Day Schedule followed by Assign Day Schedule.

Table 215 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 215 Modify Day Schedule Return Codes

Table 216 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
TimeDestinationName	The name of the Time Destination to which Day Schedules are assigned.	1..15	Character String	required
DayScheduleNameSun	The name of an existing Day Schedule to assign to the Time Destination for Sunday.	1..15	Character String	N/A
DayScheduleNameMon	The name of an existing Day Schedule to assign to the Time Destination for Monday.	1..15	Character String	N/A
DayScheduleNameTue	The name of an existing Day Schedule to assign to the Time Destination for Tuesday.	1..15	Character String	N/A
DayScheduleNameWed	The name of an existing Day Schedule to assign to the Time Destination for Wednesday.	1..15	Character String	N/A
DayScheduleNameThu	The name of an existing Day Schedule to assign to the Time Destination for Thursday.	1..15	Character String	N/A
DayScheduleNameFri	The name of an existing Day Schedule to assign to the Time Destination for Friday.	1..15	Character String	N/A
DayScheduleNameSat	The name of an existing Day Schedule to assign to the Time Destination for Saturday.	1..15	Character String	N/A

Table 216 Modify Day Schedule Input Fields

8.13.5 Remove Day Schedule

This function is used to remove a Day Schedule and all of its Period Schedules, provided it is not assigned to a Time Destination.

Note that this operation completely removes the Day Schedule; it will no longer be available for future assignment.

Table 217 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
TBD	N/A	subRC_SUCCESS

Table 217 Remove Day Schedule Return Codes

Table 218 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Name	The name of the Day Schedule to remove.	1..15	Character String	required
TimeDestinationName	The name of the Time Destination from which the Day Schedule is to be removed.	1..15	Character String	required

Table 218 Remove Day Schedule Input Fields

8.13.6 Query Day Schedule

This function is used to retrieve and display a Day Schedule. If the Name parameter is supplied, only that Day Schedule is retrieved; otherwise the Day Schedules are retrieved. Likewise, if the Time Destination parameter is specified, only the Day Schedules for that Time Destination are retrieved; if not, the complete list of existing Day Schedules is retrieved.

Table 219 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
TBD	N/A	subRC_SUCCESS

Table 219 Query Day Schedule Return Codes

Table 220 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Specific	Indicates type of query. If True, the request is for a specific DaySchedule; if False, for all DaySchedules within the specified TimeDestination.	Refer to the paragraph at the beginning of Chapter 8.		
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			
Name	The name of the Day Schedule to retrieve.	1..15	Character String	N/A

Table 220 Query Day Schedule Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
TimeDestinationName	The name of the Time Destination for which the Day Schedule(s) are to be retrieved.	1..15	Character String	N/A

Table 220 Query Day Schedule Input Fields (Sheet 2 of 2)

8.14 Period Schedule

8.14.1 Create Period Schedule

This function is used to create a Period Schedule. The Start and Stop Times for a set of Period Schedules created for a particular Day Schedule should cover the complete 24 hour day and cannot overlap. For example, PS1 can be created with StartTime 00:00 and StopTime 13:00. Then PS2 could be created for 13:01-16:00 and PS3 for 16:01-23:59. A PS2 for 12:00-16:00 would not be allowed.

Table 221 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 221 Create Period Schedule Return Codes

Table 222 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Name	The name of the Period Schedule to create.	1..15	Character String	required
DayScheduleName	The name of an existing Day Schedule.	1..15	Character String	required
TimeDestinationName	The name of the Time Destination to which the specified Day Schedule belongs.	1..15	Character String	required
StartTime	The starting time of the period in the form “HH:MM”		Character String	required
StopTime	The stop time of the period in the form “HH:MM”		Character String	required

Table 222 Create Period Schedule Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
DestinationName	The generic destination with which this period is associated.		Character String	required

Table 222 Create Period Schedule Input Fields (Sheet 2 of 2)

8.14.2 Remove Period Schedule

This function is used to remove a Period Schedule, provided its parent Day Schedule is not explicitly assigned to a Time Destination with the Assign Day Schedule command.

Table 223 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 223 Remove Period Schedule Return Codes

Table 224 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Name	The name of the Period Schedule to remove.	1..15	Character String	required
DayScheduleName	The name of the Day Schedule to which the Period Schedule belongs.	1..15	Character String	required
TimeDestinationName	The name of the Time Destination to which the Day Schedule belongs.	1..15	Character String	required
StartTime	The starting time of the period in the form “HH:MM”		Character String	required

Table 224 Remove Period Schedule Input Fields

8.14.3 Modify Period Schedule

This function is used to modify a Period Schedule. Table 225 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 225 Modify Period Schedule Return Codes

Table 226 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Name	The name of the Period Schedule to modify.	1..15	Character String	required
DayScheduleName	The name of the Day Schedule to which the Period Schedule belongs.	1..15	Character String	required
TimeDestinationName	The name of the Time Destination to which the Day Schedule belongs.	1..15	Character String	N/A
StartTime	The starting time of the period in the form “HH:MM”		Character String	N/A
StopTime	The stop time of the period in the form “HH:MM”		Character String	N/A
DestinationName	The generic destination with which this period is associated.		Character String	N/A

Table 226 Modify Period Schedule Input Fields

8.14.4 Query Period Schedule

This function is used to retrieve and display a Period Schedule. If the Time Destination is given and/or the Day Schedule is given, they act as a filter on the set of Period Schedules that are retrieved.

Note that if a specific Period Schedule object is identified by the combination of the Day Schedule and the StartTime, the Day Schedule is NOT used as a filter.

Table 227 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 227 Query Period Schedule Return Codes

Table 228 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Name	The name of the Period Schedule to remove.	1..15	Character String	N/A
DayScheduleName	The name of the Day Schedule to which the Period Schedule belongs.	1..15	Character String	N/A
TimeDestinationName	The name of the Time Destination to which the Day Schedule belongs.	1..15	Character String	N/A
StartTime	The starting time of the period in the form “HH:MM”		Character String	N/A

Table 228 Query Period Schedule Input Fields

8.15 Weekly Schedule

8.15.1 Query Weekly Schedule

This function is used to retrieve and display a Weekly Schedule. The Time Destination Name must be provided, and the day of the week may be provided as a filter option.

The response consists of an array of Weekly Schedule elements, each of which contains the Time Destination, the day of the week, and the Day Schedule Name. Note that Day Schedules may appear more than once in a Weekly Schedule, in particular when the Day Schedule is assigned to more than one day for a Time Destination.

Table 229 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 229 Query Weekly Schedule Return Codes

Table 230 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		

Table 230 Query Weekly Schedule Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
Specific	Refer to the paragraph at the beginning of Chapter 8.			
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			
TimeDestinationName	The name of an existing Time Destination.	1..15	Character String	N/A
DayOfWeek	The day of the week for which to return the schedule.	enum	DayIgnored DayUnused DayUnassigned Sun Mon Tue Wed Thu Fri Sat	N/A

Table 230 Query Weekly Schedule Input Fields (Sheet 2 of 2)

Table 231 lists the output fields for this method.

Field Name	Description	Length	Range/Values	Default Value
An array of WeeklySchedule elements, each consisting of the following attributes:				
Name	An identifier that is automatically generated by the system to identify the weekly schedule for display purposes.	1..15	Character String	N/A
Day	The day of the week for which to this schedule element.	enum	DayIgnored DayUnused DayUnassigned Sun Mon Tue Wed Thu Fri Sat	N/A
TimeDestinationName	The name of an existing Time Destination.	1..15	Character String	N/A
DayScheduleName	The name of the Day Schedule to which the Period Schedule belongs.	1..15	Character String	N/A

Table 231 Query Weekly Schedule Output Fields

8.16 Intercept

8.16.1 Create Intercept

This function is used to create an Intercept, which is a series of treatments, where each treatment is a recorded announcement, a tone, an operator, or some combination of these. The sequence of treatments can be repeated for a certain number of cycles. Intercepts are provided for calls that cannot be completed as dialed.

Table 232 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 232 Create Intercept Return Codes

Table 233 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Table 10 on page 66.		
CreateName	A flag that is set to True if only the intercept name is to be created or False if the Intercept Name already exists and this request is to provide the treatment data for it. Note that if CreateName is True, any Treatment information that is included in the request is ignored.	Boolean	False True	False
InterceptData. InterceptInfo. Name	The name of an Intercept.	1..15	Character String	required
InterceptData. Cycle	The number of times that the intercept treatment sequence is applied. This value is the same for all treatments assigned to the same intercept.	integer		1
InterceptData. Interval	The time that silence between cycles shall last. This value is the same for all treatments assigned to the same intercept.	integer		0
InterceptData. IsDefault	Indicates whether or not the intercept may be used as default.	Boolean	False True	False

Table 233 Create Intercept Input Fields (Sheet 1 of 5)

Field Name	Description	Length	Range/Values	Default Value
InterceptData. DefaultName	Identifies a default intercept entry. If a DefaultName is assigned, then all of its corresponding assigned treatments also become the treatments used, unless they are explicitly modified to be different.	1..15	Character String	required
NumberTreatments	The number of treatments in the following array.	integer	0..3	N/A
Treatment	An array of TreatmentDataInfoelements, each consisting of the following attributes:			
TreatmentDataInfo. TreatmentInfo. Name	The name of the treatment.	0..15	Character string	“”
TreatmentDataInfo. TreatmentInfo. TreatmentOrder	This parameter specifies a user-assigned value describing the desired ordering of the new treatment to be created in relation to existent treatments.	enum	TreatmentOrder Unused, TreatmentOrderFirst, TreatmentOrder Second, TreatmentOrder Third	required
TreatmentDataInfo. TreatmentInfo. InterceptId. Name	The name of an intercept.	0..15	Character string	“”
TreatmentDataInfo. Cycle	The number of times that the intercept treatment sequence is applied. This value is the same for all treatments assigned to the same intercept.	integer		required
TreatmentDataInfo. Interval	The time that silence between cycles shall last. This value is the same for all treatments assigned to the same intercept.	integer		0
TreatmentDataInfo. TreatmentType	The type of treatment. Note that if TreatmentTypeTone is specified, the contents of the Tone structure below should be provided; if TreatmentTypeAnnouncemen t is specified, the contents of the Announce structure below should be provided.	enum	TreatmentTypeUnused TreatmentTypeUnassigned TreatmentTypeTone TreatmentTypeAnnouncemen t TreatmentTypeRelCall	required

Table 233 Create Intercept Input Fields (Sheet 2 of 5)

Field Name	Description	Length	Range/Values	Default Value
TreatmentDataInfo. Tone. ToneName	Identifies a tone.	0..15	Character string	required
TreatmentDataInfo. Tone. Duration	The length of time in milliseconds for which the tone is to be played.	integer		0
TreatmentDataInfo. Tone. DestinationName	This parameter specifies the Destination name. NOTES: • If 'Destination Type' equals 'MGCPMS', the 'Destination Name' is a user-assigned alphanumeric string identifying a Media Gateway. • If 'Destination Type' equals 'DEST', the 'Destination Name' is a user-assigned alphanumeric string identifying a Destination with a 'Destination Id' of type 'MSDEST'. • This object is optional for Tone treatments, and is mandatory for Announcement treatments. • If 'Destination Type' equals 'ORIGDEST_MS', the 'Destination Name' is a user-assigned, alphanumeric string identifying an Origin Destination.	0..63	Character string	“”
TreatmentDataInfo. Tone. DestinationType	This parameter specifies the destination type.	enum	DtUnassigned DtDest DtMediaGateway DtOrigDestMs	DtUnassigned
TreatmentDataInfo. Tone. DestinationCircuit	This parameter specifies the Treatment Destination Circuit name (MG_CIRCUIT_ID of the destination). The Destination Circuit Name is a user-assigned alphanumeric string identifying a Media Gateway Circuit.	0..63	Character string	“”
TreatmentDataInfo. Announce. Name	Identifies an announcement.	0..511	Character string	required

Table 233 Create Intercept Input Fields (Sheet 3 of 5)

Field Name	Description	Length	Range/Values	Default Value
TreatmentDataInfo. Announce. Answer	This parameter specifies the answer announcement. In SIP terminology, the announcement is “early media.” Possible values are: • No - the call is treated as unanswered, and the indication is not returned. • Yes - an indication will be returned to the calling party before the announcement.	enum	YnYes YnNo YnUndefined	YnNo
TreatmentDataInfo. Announce. BargeIn	This parameter specifies whether or not the Barge-In announcement is applied. Possible values are: • No - Non-Barge-In announcement implies that a call is connected to a recorded announcement only at the beginning of the recorded message cycle. • Yes - Barge-In announcement implies that a call is connected to the recorded announcement immediately.	enum	YnYes YnNo YnUndefined	YnNo

Table 233 Create Intercept Input Fields (Sheet 4 of 5)

Field Name	Description	Length	Range/Values	Default Value
TreatmentDataInfo. Announce. DestinationName	This parameter specifies the Destination name. NOTES: • If 'Destination Type' equals 'MGCPMS', the 'Destination Name' is a user-assigned alphanumeric string identifying a Media Gateway. • If 'Destination Type' equals 'DEST', the 'Destination Name' is a user-assigned alphanumeric string identifying a Destination with a 'Destination Id' of type 'MSDEST'. • This object is optional for Tone treatments, and is mandatory for Announcement treatments. • If 'Destination Type' equals 'ORIGDEST_MS', the 'Destination Name' is a user-assigned, alphanumeric string identifying an Origin Destination.	0..63	Character string	“”
TreatmentDataInfo. Announce. DestinationType	This parameter specifies the destination type.	enum	ICDtUnused ICDtUnassigned ICDtDest ICDtMediaGateway ICDtOrigDestMs	DtUnassigned
TreatmentDataInfo. Announce. DestinationCircuit	This parameter specifies the Treatment Destination Circuit name (MG_CIRCUIT_ID of the destination). The Destination Circuit Name is a user-assigned alphanumeric string identifying a Media Gateway Circuit.	0..63	Character string	“”
SipReasonId	An optional SIP reason header that may be provided if TreatmentType is TreatmentTypeRelCall.	integer		

Table 233 Create Intercept Input Fields (Sheet 5 of 5)

8.16.2 Remove Intercept

This function is used to remove an Intercept. Table 234 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 234 Remove Intercept Return Codes

Table 235 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
RemoveName	A flag that is set to True if only the intercept name is to be removed. If RemoveName is False, the Treatment data for the Intercept is unassigned, but the Intercept itself is not deleted. If RemoveName is True, the Intercept is deleted, providing that it has no Treatment data assigned. If there is Treatment data assigned and RemoveName is True, the request is rejected.	Boolean	False True	False
InterceptInfo. Name	The name of an Intercept.	1..15	Character String	required

Table 235 Remove Intercept Input Fields

8.16.3 Modify Intercept

This function is used to modify an Intercept. Table 236 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 236 Modify Intercept Return Codes

Table 237 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
InterceptInfo. Name	The name of an Intercept.	1..15	Character String	required
NewName	The new name of an Intercept	1…15	Character String	required
Cycle	The number of times that the intercept treatment sequence is applied. This value is the same for all treatments assigned to the same intercept.	integer		1
Interval	The time that silence between cycles shall last. This value is the same for all treatments assigned to the same intercept.	integer		0
IsDefault	Indicates whether or not the intercept may be used as default.	Boolean	False True	False
DefaultName	Identifies a default intercept entry. If a DefaultName is assigned, then all of its corresponding assigned treatments also become the treatments used, unless they are explicitly modified to be different.	1..15	Character String	required

Table 237 Modify Intercept Input Fields

8.16.4 Query Intercept

This function is used to retrieve and display an Intercept. Table 238 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 238 Query Intercept Return Codes

Table 239 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Specific	Refer to the paragraph at the beginning of Chapter 8.			
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			
FilterSet	Indicates that only intercepts matching the specified filter condition should be returned. Note that FilterSetAbstract indicates those Intercepts without treatments, while FilterSetConcrete indicates those with treatments.	enum	FilterSetUnused FilterSetAll FilterSetAbstract FilterSetConcrete FilterSetSystem FilterSetUser FilterSetDefault	FilterSetAll
InterceptInfo. Name	The name of an Intercept. If provided, only the matching intercept is retried.	1..15	Character String	“”
DefaultName	Identifies a default intercept entry. If a DefaultName is assigned, then all of its corresponding assigned treatments also become the treatments used, unless they are explicitly modified to be different.	1..15	Character String	“”

Table 239 Query Intercept Input Fields

Table 240 lists the output fields for this method.

Field Name	Description	Length	Range/Values	Default Value
An array of Intercept elements, each consisting of the following attributes:				
InterceptData. InterceptInfo. Name	The name of the Intercept.	1..15	Character String	N/A
InterceptData. Cycle	The number of times that the intercept treatment sequence is applied. This value is the same for all treatments assigned to the same intercept.	integer		N/A

Table 240 Query Intercept Output Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
InterceptData. Interval	The time that silence between cycles shall last. This value is the same for all treatments assigned to the same intercept.	integer		N/A
InterceptData. IsDefault	Indicates whether or not the intercept may be used as default.	Boolean	False True	N/A
InterceptData. DefaultName	Identifies a default intercept entry. If a DefaultName is assigned, then all of its corresponding assigned treatments also become the treatments used, unless they are explicitly modified to be different.	1..15	Character String	N/A

Table 240 Query Intercept Output Fields (Sheet 2 of 2)

8.17 Treatment

8.17.1 Assign Treatment

This function is used to assign a Treatment to an existing Intercept. A treatment can be defined as a tone or announcement. Each tone is identified by a system defined tone ID and lasts for a given duration.

Table 241 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 241 Assign Treatment Return Codes

Table 242 lists the input fields for this method.

Field Name	Description	Length	Range/Value	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
TreatmentInfo. Name	The name of the treatment.	0..15	Character string	“”

Table 242 Assign Treatment Input Fields (Sheet 1 of 5)

Field Name	Description	Length	Range/Value	Default Value
TreatmentInfo. TreatmentOrder	This parameter specifies a user-assigned value describing the desired ordering of the treatment.	enum	TreatmentOrder Unused, TreatmentOrderFirst, TreatmentOrder Second, TreatmentOrder Third	required
InterceptId. Name	The name of an Intercept.	1..15	Character String	required
Cycle	The number of times that the intercept treatment sequence is applied. This value is the same for all treatments assigned to the same intercept.	integer		required
Interval	The time that silence between cycles shall last. This value is the same for all treatments assigned to the same intercept.	integer		0
TreatmentType	The type of treatment. Note that if TreatmentTypeTone is specified, the contents of the Tone structure below should be provided; if TreatmentTypeAnnouncement is specified, the contents of the Announce structure below should be provided.	enum	TreatmentTypeUnused TreatmentTypeUnassigned TreatmentTypeTone TreatmentTypeAnnouncement TreatmentTypeRelCall	required
Tone. ToneName	Identifies a tone.	0..15	Character string	required
Tone. Duration	The length of time in milliseconds for which the tone is to be played.	integer		0

Table 242 Assign Treatment Input Fields (Sheet 2 of 5)

Field Name	Description	Length	Range/Value	Default Value
Tone. DestinationName	This parameter specifies the Destination name. NOTES: • If 'Destination Type' equals 'MGCPMS', the 'Destination Name' is a user-assigned alphanumeric string identifying a Media Gateway. • If 'Destination Type' equals 'DEST', the 'Destination Name' is a user-assigned alphanumeric string identifying a Destination with a 'Destination Id' of type 'MSDEST'. • This object is optional for Tone treatments, and is mandatory for Announcement treatments. • If 'Destination Type' equals 'ORIGDEST_MS', the 'Destination Name' is a user-assigned, alphanumeric string identifying an Origin Destination.	0..63	Character string	“”
Tone. DestinationType	This parameter specifies the destination type.	enum	DtUnassigned DtDest DtMediaGateway DtOrigDestMs	DtUnassigned
Tone. DestinationCircuit	This parameter specifies the Treatment Destination Circuit name (MG_CIRCUIT_ID of the destination). The Destination Circuit Name is a user-assigned alphanumeric string identifying a Media Gateway Circuit.	0..63	Character string	“”
Announce. Name	Identifies an announcement.	0..511	Character string	required

Table 242 Assign Treatment Input Fields (Sheet 3 of 5)

Field Name	Description	Length	Range/Value	Default Value
Announce. Answer	This parameter specifies the answer announcement. In SIP terminology, the announcement is “early media.” Possible values are: • No - the call is treated as unanswered, and the indication is not returned. • Yes - an indication will be returned to the calling party before the announcement.	enum	YnYes YnNo YnUndefined	YnNo
Announce. BargeIn	This parameter specifies whether or not the Barge-In announcement is applied. Possible values are: • No - Non-Barge-In announcement implies that a call is connected to a recorded announcement only at the beginning of the recorded message cycle. • Yes - Barge-In announcement implies that a call is connected to the recorded announcement immediately.	enum	YnYes YnNo YnUndefined	YnNo

Table 242 Assign Treatment Input Fields (Sheet 4 of 5)

Field Name	Description	Length	Range/Value	Default Value
TreatmentData. Announce. DestinationName	This parameter specifies the Destination name. NOTES: • If 'Destination Type' equals 'MGCPMS', the 'Destination Name' is a user-assigned alphanumeric string identifying a Media Gateway. • If 'Destination Type' equals 'DEST', the 'Destination Name' is a user-assigned alphanumeric string identifying a Destination with a 'Destination Id' of type 'MSDEST'. • This object is optional for Tone treatments, and is mandatory for Announcement treatments. • If 'Destination Type' equals 'ORIGDEST_MS', the 'Destination Name' is a user-assigned, alphanumeric string identifying an Origin Destination.	0..63	Character string	“”
TreatmentData. Announce. DestinationType	This parameter specifies the destination type.	enum	DtUnassigned DtDest DtMediaGateway DtOrigDestMs	DtUnassigned
TreatmentData. Announce. DestinationCircuit	This parameter specifies the Treatment Destination Circuit name (MG_CIRCUIT_ID of the destination). The Destination Circuit Name is a user-assigned alphanumeric string identifying a Media Gateway Circuit.	0..63	Character string	“”
SipReasonId	An optional SIP reason header that may be provided if TreatmentType is TreatmentTypeRelCall.	integer		

Table 242 Assign Treatment Input Fields (Sheet 5 of 5)

8.17.2 Modify Treatment

This function is used to modify a Treatment. Note that TreatmentType cannot be modified. If the TreatmentType is Tone, the Tone Name and Duration may be modified. If the TreatmentType is Announce, any of the Announcement related parameters may be modified. If the TreatmentType is Relcall, only the SipReasonId may be modified.

Table 243 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 243 Modify Treatment Return Codes

Table 244 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
TreatmentInfo. Name	The name of the treatment.	0..15	Character string	“”
TreatmentInfo. TreatmentOrder	This parameter specifies a user-assigned value describing the desired ordering of the treatment.	enum	TreatmentOrder Unused, TreatmentOrderFirst, TreatmentOrder Second, TreatmentOrder Third	required
InterceptId. Name	The name of an Intercept.	1..15	Character String	required
Cycle	The number of times that the intercept treatment sequence is applied. This value is the same for all treatments assigned to the same intercept.	integer		required
Interval	The time that silence between cycles shall last. This value is the same for all treatments assigned to the same intercept.	integer		0
Tone. ToneName	Identifies a tone.	0..15	Character string	required

Table 244 Modify Treatment Input Fields (Sheet 1 of 3)

Field Name	Description	Length	Range/Values	Default Value
Tone. Duration	The length of time in milliseconds for which the tone is to be played.	integer		0
Announce. Name	Identifies an announcement.	0..511	Character string	required
Announce. Answer	This parameter specifies the answer announcement. In SIP terminology, the announcement is “early media.” Possible values are: • No - the call is treated as unanswered, and the indication is not returned. • Yes - an indication will be returned to the calling party before the announcement.	enum	YnYes YnNo YnUndefined	YnNo
Announce. BargeIn	This parameter specifies whether or not the Barge-In announcement is applied. Possible values are: • No - Non-Barge-In announcement implies that a call is connected to a recorded announcement only at the beginning of the recorded message cycle. • Yes - Barge-In announcement implies that a call is connected to the recorded announcement immediately.	enum	YnYes YnNo YnUndefined	YnNo

Table 244 Modify Treatment Input Fields (Sheet 2 of 3)

Field Name	Description	Length	Range/Values	Default Value
TreatmentData. Announce. DestinationName	This parameter specifies the Destination name. NOTES: • If 'Destination Type' equals 'MGCPMS', the 'Destination Name' is a user-assigned alphanumeric string identifying a Media Gateway. • If 'Destination Type' equals 'DEST', the 'Destination Name' is a user-assigned alphanumeric string identifying a Destination with a 'Destination Id' of type 'MSDEST'. • This object is optional for Tone treatments, and is mandatory for Announcement treatments. • If 'Destination Type' equals 'ORIGDEST_MS', the 'Destination Name' is a user-assigned, alphanumeric string identifying an Origin Destination.	0..63	Character string	“”
TreatmentData. Announce. DestinationType	This parameter specifies the destination type.	enum	DtUnassigned DtDest DtMediaGateway DtOrigDestMs	DtUnassigned
TreatmentData. Announce. DestinationCircuit	This parameter specifies the Treatment Destination Circuit name (MG_CIRCUIT_ID of the destination). The Destination Circuit Name is a user-assigned alphanumeric string identifying a Media Gateway Circuit.	0..63	Character string	“”
SipReasonId	An optional SIP reason header that may be provided if TreatmentType is TreatmentTypeRelCall.	integer		

Table 244 Modify Treatment Input Fields (Sheet 3 of 3)

8.17.3 Unassign Treatment

This function is used to unassign a Treatment from an existing Intercept. If no remaining treatments are assigned to the Intercept after the Unassign operation, the Cycle and Interval parameters are reset to default values.

Table 245 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 245 Unassign Treatment Return Codes

Table 246 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
TreatmentInfo. Name	The name of the treatment.	0..15	Character string	required
TreatmentInfo. TreatmentOrder	This parameter specifies the ordering of the treatment.	enum	TreatmentOrder Unused, TreatmentOrderFirst, TreatmentOrder Second, TreatmentOrder Third	required
InterceptId. Name	The name of the Intercept from which the Treatment is to be unassigned.	1..15	Character String	required
TreatmentData. Announce. DestinationName	This parameter specifies the Destination name. NOTES: • If 'Destination Type' equals 'MGCPMS', the 'Destination Name' is a user-assigned alphanumeric string identifying a Media Gateway. • If 'Destination Type' equals 'DEST', the 'Destination Name' is a user-assigned alphanumeric string identifying a Destination with a 'Destination Id' of type 'MSDEST'. • This object is optional for Tone treatments, and is mandatory for Announcement treatments. • If 'Destination Type' equals 'ORIGDEST_MS', the 'Destination Name' is a user-assigned, alphanumeric string identifying an Origin Destination.	0..63	Character string	“”

Table 246 Unassign Treatment Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
TreatmentData. Announce. DestinationType	This parameter specifies the destination type.	enum	DtUnassigned DtDest DtMediaGateway DtOrigDestMs	DtUnassigned
TreatmentData. Announce. DestinationCircuit	This parameter specifies the Treatment Destination Circuit name (MG_CIRCUIT_ID of the destination). The Destination Circuit Name is a user-assigned alphanumeric string identifying a Media Gateway Circuit.	0..63	Character string	“”
SipReasonId	An optional SIP reason header that may be provided if TreatmentType is TreatmentTypeRelCall.	integer		

Table 246 Unassign Treatment Input Fields (Sheet 2 of 2)

8.17.4 Query Treatment

This function is used to retrieve and display existing Treatments. Table 247 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 247 Query Treatment Return Codes

Table 248 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Specific	Refer to the paragraph at the beginning of Chapter 8.			
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			
InterceptInfo. Name	The name of an Intercept. If provided, only the matching intercept is retried.	1..15	Character String	“”
TreatmentInfo. Name	The name of the treatment.	0..15	Character string	“”

Table 248 Query Treatment Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
TreatmentInfo. TreatmentOrder	This parameter specifies the ordering of the treatment.	enum	TreatmentOrder Unused, TreatmentOrderFirst, TreatmentOrder Second, TreatmentOrder Third	N/A

Table 248 Query Treatment Input Fields (Sheet 2 of 2)

Table 249 lists the output fields for this method.

Field Name	Description	Length	Range/Values	Default Value
Treatment	An array of TreatmentData elements, each consisting of the following attributes:			
TreatmentData. TreatmentInfo. Name	The name of the treatment.	0..15	Character string	“”
TreatmentData. TreatmentInfo. TreatmentOrder	This parameter specifies a user-assigned value describing the desired ordering of the new treatment to be created in relation to existent treatments.	enum	TreatmentOrder Unused, TreatmentOrderFirst, TreatmentOrder Second, TreatmentOrder Third	required
TreatmentData. TreatmentInfo. InterceptId. Name	The name of an intercept.	0..15	Character string	“”
TreatmentData. Cycle	The number of times that the intercept treatment sequence is applied. This value is the same for all treatments assigned to the same intercept.	integer		required
TreatmentData. Interval	The time that silence between cycles shall last. This value is the same for all treatments assigned to the same intercept.	integer		0

Table 249 Query Treatment Output Fields (Sheet 1 of 4)

Field Name	Description	Length	Range/Values	Default Value
TreatmentData. TreatmentType	The type of treatment. Note that if TreatmentTypeTone is specified, the contents of the Tone structure below should be provided; if TreatmentTypeAnnouncement is specified, the contents of the Announce structure below should be provided.	enum	TreatmentTypeUnused TreatmentTypeUnassigned TreatmentTypeTone TreatmentTypeAnnouncement TreatmentTypeRelCall	required
TreatmentData. Tone. ToneName	Identifies a tone.	0..15	Character string	required
TreatmentData. Tone. Duration	The length of time in milliseconds for which the tone is to be played.	integer		0
TreatmentData. Tone. DestinationName	This parameter specifies the Destination name. NOTES: • If 'Destination Type' equals 'MGCPMS', the 'Destination Name' is a user-assigned alphanumeric string identifying a Media Gateway. • If 'Destination Type' equals 'DEST', the 'Destination Name' is a user-assigned alphanumeric string identifying a Destination with a 'Destination Id' of type 'MSDEST'. • This object is optional for Tone treatments, and is mandatory for Announcement treatments. • If 'Destination Type' equals 'ORIGDEST_MS', the 'Destination Name' is a user-assigned, alphanumeric string identifying an Origin Destination.	0..63	Character string	“”
TreatmentData. Tone. DestinationType	This parameter specifies the destination type.	enum	DtUnassigned DtDest DtMediaGateway DtOrigDestMs	DtUnassigned

Table 249 Query Treatment Output Fields (Sheet 2 of 4)

Field Name	Description	Length	Range/Values	Default Value
TreatmentData. Tone. DestinationCircuit	This parameter specifies the Treatment Destination Circuit name (MG_CIRCUIT_ID of the destination). The Destination Circuit Name is a user-assigned alphanumeric string identifying a Media Gateway Circuit.	0..63	Character string	“”
TreatmentData. Announce. Name	Identifies an announcement.	0..511	Character string	required
TreatmentData. Announce. Answer	This parameter specifies the answer announcement. In SIP terminology, the announcement is “early media.” Possible values are: • No - the call is treated as unanswered, and the indication is not returned. • Yes - an indication will be returned to the calling party before the announcement.	enum	YnYes YnNo YnUndefined	YnNo
TreatmentData. Announce. BargeIn	This parameter specifies whether or not the Barge-In announcement is applied. Possible values are: • No - Non-Barge-In announcement implies that a call is connected to a recorded announcement only at the beginning of the recorded message cycle. • Yes - Barge-In announcement implies that a call is connected to the recorded announcement immediately.	enum	YnYes YnNo YnUndefined	YnNo
TreatmentData. Announce. DestinationType	This parameter specifies the destination type.	enum	DtUnassigned DtDest DtMediaGateway DtOrigDestMs	DtUnassigned

Table 249 Query Treatment Output Fields (Sheet 3 of 4)

Field Name	Description	Length	Range/Values	Default Value
TreatmentData. Announce. DestinationName	This parameter specifies the Destination name. NOTES: • If 'Destination Type' equals 'MGCPMS', the 'Destination Name' is a user-assigned alphanumeric string identifying a Media Gateway. • If 'Destination Type' equals 'DEST', the 'Destination Name' is a user-assigned alphanumeric string identifying a Destination with a 'Destination Id' of type 'MSDEST'. • This object is optional for Tone treatments, and is mandatory for Announcement treatments. • If 'Destination Type' equals 'ORIGDEST_MS', the 'Destination Name' is a user-assigned, alphanumeric string identifying an Origin Destination.	0..63	Character string	“”
TreatmentData. Announce. DestinationCircuit	This parameter specifies the Treatment Destination Circuit name (MG_CIRCUIT_ID of the destination). The Destination Circuit Name is a user-assigned alphanumeric string identifying a Media Gateway Circuit.	0..63	Character string	“”
SipReasonId	An optional SIP reason header that may be provided if TreatmentType is TreatmentTypeRelCall.	integer		

Table 249 Query Treatment Output Fields (Sheet 4 of 4)

8.17.5 Prioritize Treatment

This function is used to change the ordered priority of a Treatment. Table 250 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 250 Prioritize Treatment Return Codes

Table 251 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
NewOrder	This parameter specifies the desired ordering of the treatment.	enum	TreatmentOrder Unused, TreatmentOrderFirst, TreatmentOrder Second, TreatmentOrder Third	required
TreatmentInfo. Name	The name of the treatment.	0..15	Character string	“““
TreatmentInfo. TreatmentOrder	This parameter specifies the existing ordering of the treatment whose order is to be changed.	enum	TreatmentOrder Unused, TreatmentOrderFirst, TreatmentOrder Second, TreatmentOrder Third	required
InterceptId. Name	The name of an Intercept.	1..15	Character String	required

Table 251 Prioritize Treatment Input Fields

8.18 SIP Trusted Entity Ports

8.18.1 Add SIP Trusted Entity Ports

This function is used to add port ranges to a SIP Entity.

A list of port ranges may be provided in this request, and there may be cases in which some port ranges cannot be successfully assigned to the given SIP Entity. In this case, those ports are skipped and processing continues in an attempt to assign the remaining port ranges in the list provided. The response to the operator includes a list of port ranges that could not be added along with an error message indicating the reason for the failure.

Table 252 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 252 Add SIP Trusted Entity Ports Return Codes

Table 253 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
SipEntityIPAddress	The IP address of the SIP Entity.	7..16	A valid IP address.	required
PortRangeList	An array of PortRangeEntry elements, each consisting of the following attributes:			
minPort	The lower end of the port range to add. If minPort=0 then the entire range of ports for the specified IP address is defined as trusted.	integer	0..65535	required
maxPort	The upper end of the port range to add. If maxPort=0 then only the single port specified by minPort is defined as trusted.	integer	0..65535	required

Table 253 Add SIP Trusted Entity Ports Input Fields

8.18.2 Remove SIP Trusted Entity Ports

This function is used to remove port ranges from a SIP Entity.

A list of port ranges may be provided in this request, and there may be cases in which some port ranges cannot be successfully removed from the given SIP Entity. In this case, those ports are skipped and processing continues in an attempt to remove the remaining port ranges in the list provided. The response to the operator includes a list of port ranges that could not be removed along with an error message indicating the reason for the failure.

Table 254 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 254 Remove SIP Trusted Entity Ports Return Codes

Table 255 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
SipEntityIPAddress	The IP address of the SIP Entity.	7..16	A valid IP address.	required
PortRangeList	An array of PortRangeEntry elements, each consisting of the following attributes:			
minPort	The lower end of the port range to add. If minPort=0 then the entire range of ports for the specified IP address is defined as trusted.	integer	0..65535	required
maxPort	The upper end of the port range to add. If maxPort=0 then only the single port specified by minPort is defined as trusted.	integer	0..65535	required

Table 255 Remove SIP Trusted Entity Ports Input Fields

8.18.3 Get SIP Trusted Entity Ports List

This function is used to retrieve and display the list of port ranges assigned to a SIP Entity.

If the IP address of a SIP Entity is provided, the list of port ranges that are assigned to this entity will be displayed. If the IP address is omitted, the entire list of existing SIP Entities will be displayed along with all of their corresponding port ranges.

Table 256 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 256 Get SIP Trusted Entity Ports List Return Codes

Table 257 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		

Table 257 Get SIP Trusted Entity Ports List Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			
SipEntityIPAddress	The IP address of the SIP Entity.	7..16	A valid IP address.	required
PortRangeList	An array of PortRangeEntry elements, each consisting of the following attributes:			
minPort	The lower end of the port range to add. If minPort=0 then the entire range of ports for the specified IP address is defined as trusted.	integer	0..65535	required
maxPort	The upper end of the port range to add. If maxPort=0 then only the single port specified by minPort is defined as trusted.	integer	0..65535	required

Table 257 Get SIP Trusted Entity Ports List Input Fields (Sheet 2 of 2)

Table 258 lists the output fields for this method.

Field Name	Description	Length	Range/Values	Default Value
PortRangeListData	An array of PortRangeListDataEntry elements, each consisting of the following attributes:			
SipEntityIPAddress	The IP address of the SIP Entity.	7..16	A valid IP address.	required
minPort	The lower end of the port range to add. If minPort=0 then the entire range of ports for the specified IP address is defined as trusted.	integer	0..65535	required
maxPort	The upper end of the port range to add. If maxPort=0 then only the single port specified by minPort is defined as trusted.	integer	0..65535	required

Table 258 Get SIP Trusted Entity Ports Lists Output Fields

8.19 System SIP Entry

8.19.1 Add System SIP Entry

This function is used to create an entry in the system SIP security table.

Table 259 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 259 Add System SIP Entry Return Codes

Table 260 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
SipEntityID	This identifies the HTTP Digest Authentication entry; typically it is an IP address or a Fully Qualified Domain Name (FQDN).	0..255	A character string representing a valid IP address or FQDN.	required
RealmIDType	This identifies the HTTP Digest identification type being used.	enum	Ip fqdn	Ip
TrustedEntity	Indicates whether the SystemSipEntry being created is to be a trusted entity or not.	boolean	false, true	false
Scheme	The type of security scheme that is being implemented. In this release, only the digest_authentication form of security is supported.	enum	digest-authentication	digest-authentication
LocalRealm	This identifies the local realm. A realm is a string to be displayed to users for association with its username and password. For a trusted entity, this parameter should not be provisioned. For a non-trusted entity, if this parameter is not provisioned, the default localhost entry information is used.	0..128	Character string	“” OR if subscriber is OSMO, then it is assigned the LocalRealm value of the DeskPhone

Table 260 Add System SIP Entry Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
LocalUserName	This identifies a user within the local realm. For a trusted entity, this parameter should not be provisioned. For a non-trusted entity, if this parameter is not provisioned, the default localhost entry information is used.	0, 6..64	Character string	“” OR if subscriber is OSMO, then it is assigned the OSMO Subscriber DN value
LocalPassword	The password for the local user within the local realm. For a trusted entity, this parameter should not be provisioned. For a non-trusted entity, if this parameter is not provisioned, the default localhost entry information is used.	6..20	Character string	“” OR if subscriber is OSMO, then it is assigned the LocalPassword value of the DeskPhone
RemoteRealm	This identifies the remote realm. A realm is a string to be displayed to users for association with its username and password. For a trusted entity, this parameter should not be provisioned. For a non-trusted entity, this parameter must be provisioned.	0..128	Character string	“”
RemoteUserName	This identifies a user within the remote realm. For a trusted entity, this parameter should not be provisioned. For a non-trusted entity, this parameter must be provisioned.	0, 6..64	Character string	“”
RemotePassword	The password for the remote user within the remote realm. For a trusted entity, this parameter should not be provisioned. For a non-trusted entity, this parameter must be provisioned.	6..20	Character string	“”

Table 260 Add System SIP Entry Input Fields (Sheet 2 of 2)

8.19.2 Modify System SIP Entry

This function is used to modify an entry in the system SIP security table.

Table 261 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 261 Modify System SIP Entry Return Codes

Table 262 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
SipEntityID	This identifies the HTTP Digest Authentication entry; typically it is an IP address or a Fully Qualified Domain Name (FQDN).	0..255	A character string representing a valid IP address or FQDN.	required
TrustedEntity	Indicates whether the SystemSipEntry being created is to be a trusted entity or not.	boolean	false, true	N/A
Scheme	The type of security scheme that is being implemented. In this release, only the digest_authentication form of security is supported.	enum	digest-authentication	N/A
LocalRealm	This identifies the local realm. A realm is a string to be displayed to users for association with its username and password. For a trusted entity, this parameter should not be provisioned. For a non-trusted entity, if this parameter is not provisioned, the default localhost entry information is used.	0..128	Character string	N/A
LocalUserName	This identifies a user within the local realm. For a trusted entity, this parameter should not be provisioned. For a non-trusted entity, if this parameter is not provisioned, the default localhost entry information is used.	0, 6..64	Character string	N/A

Table 262 Modify System SIP Entry Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
LocalPassword	The password for the local user within the local realm. For a trusted entity, this parameter should not be provisioned. For a non-trusted entity, if this parameter is not provisioned, the default localhost entry information is used.	6..20	Character string	N/A
RemoteRealm	This identifies the remote realm. A realm is a string to be displayed to users for association with its username and password. Note that if the TrustedEntity parameter is being modified to false, this parameter must be provided.	0..128	Character string	N/A
RemoteUserName	This identifies a user within the remote realm. Note that if the TrustedEntity parameter is being modified to false, this parameter must be provided.	0, 6..64	Character string	N/A
RemotePassword	The password for the remote user within the remote realm. Note that if the TrustedEntity parameter is being modified to false, this parameter must be provided.	6..20	Character string	N/A

Table 262 Modify System SIP Entry Input Fields (Sheet 2 of 2)

8.19.3 Delete System SIP Entry

This function is used to delete an entry from the system SIP security table.

Table 263 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 263 Delete System SIP Entry Return Codes

Table 264 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		

Table 264 Delete System SIP Entry Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
SipEntityID	This identifies the HTTP Digest Authentication entry; typically it is an IP address or a Fully Qualified Domain Name (FQDN).	0..255	A character string representing a valid IP address or FQDN.	required

Table 264 Delete System SIP Entry Input Fields (Sheet 2 of 2)

8.19.4 Get System SIP Entry Info

This function is used to retrieve and display a single system SIP security entry. Table 265 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 265 Get System SIP Entry Info Return Codes

Table 266 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
SipEntityIPAddress	The IP address of the SIP Entity.	7..16	A valid IP address.	required

Table 266 Get System SIP Entry Info Input Fields

Table 267 lists the output fields for this method.

Field Name	Description	Length	Range/Values	Default Value
SipEntityID	This identifies the HTTP Digest Authentication entry; typically it is an IP address or a Fully Qualified Domain Name (FQDN).	0..255	A character string representing a valid IP address or FQDN.	N/A
RealmIDType	This identifies the HTTP Digest identification type being used.	enum	Ip fqdn	N/A
TrustedEntity	Indicates whether the SystemSipEntry being created is to be a trusted entity or not.	boolean	false, true	N/A

Table 267 Get System SIP Entry Info Output Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
Scheme	The type of security scheme that is being implemented. In this release, only the digest_authentication form of security is supported.	enum	digest-authentication	N/A
LocalRealm	This identifies the local realm. A realm is a string to be displayed to users for association with its username and password. For a trusted entity, this parameter should not be provisioned. For a non-trusted entity, if this parameter is not provisioned, the default localhost entry information is used.	0..128	Character string	N/A
LocalUserName	This identifies a user within the local realm. For a trusted entity, this parameter should not be provisioned. For a non-trusted entity, if this parameter is not provisioned, the default localhost entry information is used.	0, 6..64	Character string	N/A
LocalPassword	The password for the local user within the local realm. For a trusted entity, this parameter should not be provisioned. For a non-trusted entity, if this parameter is not provisioned, the default localhost entry information is used.	6..20	Character string	N/A
RemoteRealm	This identifies the remote realm. A realm is a string to be displayed to users for association with its username and password. For a trusted entity, this parameter should not be provisioned. For a non-trusted entity, this parameter must be provisioned.	0..128	Character string	N/A
RemoteUserName	This identifies a user within the remote realm.	0, 6..64	Character string	N/A
RemotePassword	The password for the remote user within the remote realm.	6..20	Character string	N/A

Table 267 Get System SIP Entry Info Output Fields (Sheet 2 of 2)

8.19.5 Get System SIP Entry List

This function is used to retrieve and display the list of existing system SIP security entries.

The result data contains a list of SIP Entity IDs. For detailed information about one of the SIP Entity IDs in the list, the GetSystemSIPEntryInfo operation can be used.

Table 268 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 268 Get System SIP Entry List Return Codes

Table 269 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			

Table 269 Get System SIP Entry List Input Fields

8.20 Stat Counter

8.20.1 Get Stat Counter Info

This function is used to retrieve and display a single RTP Statistics Counter, including its value and status. Table 270 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 270 Get Stat Counter Info Return Codes

Table 271 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
CounterName	The name of the RTP Statistics Counter to retrieve.	1..32	Character string that may contain only letters, digits, and underscores.	required

Table 271 Get Stat Counter Info Input Fields

Table 272 lists the output fields for this method.

Field Name	Description	Length	Range/Values	Default Value
CounterName	The name of the RTP Statistics Counter.	1..32	Character string that may contain only letters, digits, and underscores.	N/A
CounterValue	The value of the RTP Statistics Counter.	integer		N/A
CounterStatus	An indication of whether the RTP Statistics Counter is active or not.	Boolean	False, True	N/A

Table 272 Get Stat Counter Info Output Fields

8.20.2 Get Stat Counter List

This function is used to retrieve and display the list of all RTP Statistics Counters.

The result data contains a list of RTP Statistics Counter names. For detailed information about one of the RTP Statistics Counters in the list, the GetStatCounterInfo operation can be used.

Table 273 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 273 Get Stat Counter List Return Codes

Table 274 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		

Table 274 Get Stat Counter List Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			

Table 274 Get Stat Counter List Input Fields (Sheet 2 of 2)

8.21 Database Job

8.21.1 Create Database Job

This function is used to create a job to initiate a database backup or restore.

The response to this request includes a JobId that has been assigned to the request and can be used later in order to cancel or query this specific job.

Table 275 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 275 Create Database Job Return Codes

Table 276 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
DatabaseJobName	Indicates the requested action.	enum	backup restore	required
DatabaseEntity	The name of the database component to which the job applies.	enum	all oracle_rtp oracle_src oracle_all omni solid	required

Table 276 Create Database Job Input Fields

8.21.2 Cancel Database Job

This function is used to cancel an existing database backup or restore job. Table 277 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 277 Cancel Database Job Return Codes

Table 278 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
JobId	The ID that identifies the job to be canceled.	integer	A valid JobId.	required

Table 278 Cancel Database Job Input Fields

8.21.3 Delete Database Job

This function is used to delete an existing database backup or restore job. Table 279 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 279 Delete Database Job Return Codes

Table 280 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
JobId	The ID that identifies the job to be canceled.	integer	A valid JobId.	required

Table 280 Delete Database Job Input Fields

8.21.4 Get Database Job List

This function is used to retrieve and display an existing database backup or restore job, or a list of such jobs.

If the JobId is provided, the information about only that job is returned; otherwise, information about the complete list of existing database jobs is returned.

Table 281 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 281 Get Database Job List Return Codes

Table 282 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			
JobId	The ID that identifies the job to be retrieved.	integer	A valid JobId.	0

Table 282 Get Database Job List Input Fields

Table 283 lists the output fields for this method.

Field Name	Description	Length	Range/Values	Default Value
JobId	The ID that identifies the job.	integer	A valid JobId.	N/A
DatabaseJobName	Indicates the requested action.	enum	backup restore	required
ExecPath	Indicates the path to the job’s executable.		Character string	N/A
Type	A bitmask that describes the job attributes. The type may contain one or more attributes such as: BASIC (no attributes) CANCELABLE (job may be canceled) SCHEDULABLE (job may be scheduled) BLOCK_ON_DONE (job is blocked when completed) BLOCK_ON_ERR (job is blocked when an error occurs) TRAP_ON_ERR (trap is sent when job ends in error) WRITES_LOG (log file is written upon completion of the job)		Character string	N/A

Table 283 Get Database Job List Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
AvgRunTime	Indicates the estimated average run time (in seconds) required for the job to complete.		Character string	N/A
MaxRunTime	Indicates the estimated maximum run time (in seconds) required for the job to complete.		Character string	N/A
State	Indicates whether the job is: running overdue cancelling cancelled done error		Character string	N/A
ReportName	Indicates the path and name of the report generated for this job. Information contain in a log is brief.		Character string	N/A
LogName	Indicates the path and name of the log generated for this job. Information contain in a log is detailed. Not currently supported.		Character string	N/A
StartTime	Indicates the start time of the job.		Character string	N/A
StopTime	Indicates the stop time of the job.		Character string	N/A
DatabaseEntity	The name of the database component to which the job applies.	enum	all oracle_rtp oracle_src oracle_all omni solid	required

Table 283 Get Database Job List Input Fields (Sheet 2 of 2)

8.22 Calling Location

8.22.1 Create Calling Location

This function is used to create a Calling Location. Table 284 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 284 Create Calling Location Return Fields

Table 285 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Code	The calling location code.		Character String	required
Name	The calling location name.		Character String	required

Table 285 Create Calling Location Input Fields

8.22.2 Modify Calling Location

This function is used to to modify the name of a Calling Location.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 286 Modify Calling Location Return Fields

Table 287 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Code	The calling location code.		Character String	required
Name	The calling location name.		Character String	required

Table 287 Modify Calling Location Input Fields

8.22.3 Remove Calling Location

This function is used to delete an existing Calling Location.Table 288 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 288 Remove Calling Location Return Fields

Table 289 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Code	The calling location code.		Character String	required

Table 289 Remove Calling Location Input Fields

8.22.4 Query Calling Location

This function is used to retrieve and display existing Calling Locations.Table 290 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 290 Query Calling Location Return Fields

Table 291 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Specific	Indicates whether a specific Calling Location is being requested, or a list of Calling Locations.	Please refer to the paragraph at the beginning of Chapter 8.		
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			
Code	The calling location code.		Character String	N/A

Table 291 Remove Calling Location Input Fields

8.23 Origin Destination

8.23.1 Create Origin Destination

This function is used to create an Origin Destination. Table 292 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 292 Create Origin Destination Return Fields

Table 293 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Name	The name of the origin destination.		Character String	required
Type	The type of origin destination.	Enum	OrigDestCarrier OrigDestMs OrigDestUndefined OrigDestIgnored	required

Table 293 Create Origin Destination Input Fields

8.23.2 Modify Origin Destination

This function is used to modify an Origin Destination.Table 294 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 294 Modify Origin Destination Return Fields

Table 295 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		

Table 295 Modify Origin Destination Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
Name	The name of the origin destination to modify.		Character String	required
Type	The type of origin destination.	enum	OrigDestCarrier OrigDestMs OrigDestUndefined OrigDestIgnored	N/A

Table 295 Modify Origin Destination Input Fields (Sheet 2 of 2)

8.23.3 Remove Origin Destination

This function is used to delete an existing Origin Destination. Table 296 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 296 Remove Origin Destination Return Fields

Table 297 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Name	The name of the origin destination to remove.		Character String	required

Table 297 Remove Origin Destination Input Fields

8.23.4 Query Origin Destination

This function is used to retrieve and display existing Origin Destinations.Table 298 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 298 Query Origin Destination Return Fields

Table 299 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Specific	Indicates whether a specific Calling Location is being requested, or a list of Calling Locations.	Please refer to the paragraph at the beginning of Chapter 8.		
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			
Name	The name of the origin destination to retrieve.		Character String	N/A

Table 299 Query Origin Destination Input Fields

8.24 Origin Route

8.24.1 Create Origin Route

This function is used to create an Origin Route. Table 300 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 300 Create Origin Route Return Fields

Table 301 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
OriginDestinationName	The name of the origin destination with which to associate the route.		Character String	required
RateAreaName	The name of a Rate Area with which to associate the route.		Character String	required
DestinationName	The name of the destination to which calls will be translated.		Character String	required

Table 301 Create Origin Route Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
OriginDestinationType	The type of origin destination.	enum	OriginRouteDtIgnored OriginRouteDtUnused OriginRouteDtDest OriginRouteDtMediaGateway	OriginRouteDtDest

Table 301 Create Origin Route Input Fields (Sheet 2 of 2)

8.24.2 Modify Origin Route

This function is used to modify an Origin Route.Table 302 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 302 Modify Origin Route Return Fields

Table 303 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
OriginDestinationName	The name of the origin destination associated with the route.		Character String	required
RateAreaName	The name of the associated Rate Area.		Character String	N/A
DestinationName	The name of the destination to which calls will be translated.		Character String	N/A
OriginDestinationType	The type of origin destination.	enum	OriginRouteDtIgnored OriginRouteDtUnused OriginRouteDtDest OriginRouteDtMediaGateway	OriginRouteDtUnused

Table 303 Modify Origin Route Input Fields

8.24.3 Remove Origin Route

This function is used to delete an existing Origin Route.Table 304 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 304 Remove Origin Route Return Fields

Table 305 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
OriginDestinationName	The name of the origin destination to remove.		Character String	required
RateAreaName	The name of the associated Rate Area.		Character String	N/A

Table 305 Remove Origin Route Input Fields

8.24.4 Query Origin Route

This function is used to retrieve and display existing Origin Routes.Table 306 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 306 Query Origin Route Return Fields

Table 307 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Specific	Indicates whether a specific Calling Location is being requested, or a list of Calling Locations.	Please refer to the paragraph at the beginning of Chapter 8.		
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			
OriginDestinationName	The name of the origin destination associated with the route.		Character String	N/A
RateAreaName	The name of the associated Rate Area.		Character String	N/A
DestinationName	The name of the destination to which calls will be translated.		Character String	N/A
OriginDestinationType	The type of origin destination.	enum	OriginRouteDtIgnored OriginRouteDtUnused OriginRouteDtDest OriginRouteDtMediaGateway	N/A

Table 307 Query Origin Route Input Fields

8.25 Get Generic Object Attributes

Given an object name, this query returns a list of valid attributes for that object.Table 308 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS

Table 308 Get Generic Object Attributes Return Fields

Table 309 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		

Table 309 Get Generic Object Attributes Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
ObjectName The list of supported objects is as follows: E164_DESTINATION SUBSCRIBER PREFIX_ACCESS_CODE MLHG SUBSCRIBER_SIP_PROFILE BUSINESS_GROUP COS DESTINATION E164_CODE E164_DN E164_IP_DESTINATION E164_ROUTE ENDPOINT NUMBER_PLAN PNP_LOCATION_CODE PNP_LOCATION_EXTENSION RATE_AREA FEATURE ROUTE CALLING_LOCATION DEPARTMENT BG_ATTENDANT_NUM SUBSCRIBER_BG_DESTINATION PAC_DESTINATION SUBSCRIBER_BG_ENDPOINT SUBSCRIBER_KEYSET_INFO NUMBER_PLAN_EP	The object for which the list of attributes is required.	1..64	Character String representing a supported object.	required

Table 309 Get Generic Object Attributes Input Fields (Sheet 2 of 2)

Table 310 lists the output fields for this method.

Field Name	Description	Length	Range/Values	Default Value
TableRowDescriptions	This is an array of elements, each of which describes one attribute of the given object and has the following fields:			
TableRowDescription RowAttributeName	The name of the attribute.		Character String	N/A
TableRowDescription RowAttributeType	The type of the attribute.	enum	attrString_ attrDate_ attrNumber_ attrUnsigned_ attrFloat_	N/A

Table 310 Get Generic Object Attributes Output Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
TableRowDescription RowAttributeLength	The maximum length of the attribute.	integer		N/A

Table 310 Get Generic Object Attributes Output Fields (Sheet 2 of 2)

8.26 Generic Query

Given an object name and a list of attributes, this query returns the values of those attributes for that object. Optionally, conditions may be applied so that the data is only returned for those objects that match the conditions.

Note: by default all the conditions in the array of ConditionObjects are joined with an “AND”. The parameter ConditionRelationString is provided in order to allow conditions to be joined with “AND”, “AND NOT”, or “OR”, as well as to allow condition to be grouped in parentheses. Each GenericConditionObject is represented in the ConditionRelationString by its ConditionName, so if ConditionRelationString is specified in a query, each GenericConditionObject should have a ConditionName specified. Also note that the CondtionNames and the join strings (AND/AND NOT/OR) should be separated by spaces in the ConditionRelationString.

Table 311 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.02.03.02	N/A	subRC_SUCCESS subRC_APP_DSA_GENERIC_QUERY_DB_ERROR subRC_APP_DSA_GENERIC_QUERY_INIT_ERROR subRC_APP_DSA_GENERIC_QUERY_UNKNOWN_OBJECT subRC_APP_DSA_GENERIC_QUERY_UNKNOWN_ATTRIBUTE subRC_APP_DSA_GENERIC_QUERY_UNKNOWN_CONDITION_ATTRIBUTE subRC_APP_DSA_GENERIC_QUERY_ILLEGAL_OBJECT_NAME subRC_APP_DSA_GENERIC_QUERY_ILLEGAL_CONDITION subRC_APP_DSA_GENERIC_QUERY_ILLEGAL_CONDITION_VALUE subRC_APP_DSA_GENERIC_QUERY_ILLEGAL_RELATION_STRING

Table 311 Generic Query Return Fields

Table 312 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		

Table 312 Generic Query Input Fields (Sheet 1 of 4)

Field Name	Description	Length	Range/Values	Default Value
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			
ObjectName	The object for which the query is to be performed.	1..64	Character String representing a supported object. The list of supported objects is as follows: E164_DESTINATION SUBSCRIBER PREFIX_ACCESS_CODE MLHG SUBSCRIBER_SIP_PROFIL E BUSINESS_GROUP COS DESTINATION E164_CODE E164_DN E164_IP_DESTINATION E164_ROUTE ENDPOINT NUMBER_PLAN PNP_LOCATION_CODE PNP_LOCATION_EXTENSIO N RATE_AREA FEATURE ROUTE CALLING_LOCATION BG_DEPARTMENT BG_NAME BG_ATTENDANT_NUM SUBSCRIBER_BG_DESTIN ATION PAC_DESTINATION SUBSCRIBER_BG_ENDPOI NT SUBSCRIBER_KEYSET_INF O NUMBER_PLAN_EP	required

Table 312 Generic Query Input Fields (Sheet 2 of 4)

Field Name	Description	Length	Range/Values	Default Value
AttributeNames	An array of attributes in the object whose value is to be retrieved and displayed. This is an optional parameter, and the default behavior if no AttributeNames are provided is to return the values of all attributes.	1..64	Character String representing a valid object attribute. A list of all currently valid attribute names for a supported object can be obtained using the GetGenericObjectAttributes request (see Section 8.25). Also note, a special attribute $COUNT is available. If this attribute name is specified, the query just returns the number of elements.	(all)
OrderByAttribute	This is an optional parameter that the query results are to be sorted by the specified attribute. If not specified the order of the result will be as present in the database table.	1..64	Character String representing a valid object attribute. A list of all currently valid attribute names for a supported object can be obtained using the GetGenericObjectAttributes request (see Section 8.25).	N/A
ResultsInAscendingOrder	This parameter applies only if the OrderByAttribute parameter is provided, and indicates whether the results are to be sorted in ascending or descending order. The default is True (ascending order).	Boolean	True False	True
ConditionObjects	This is an array of GenericConditionObject elements and is used to specify the query conditions. This is an optional parameter. If no ConditionObject is specified, all rows for the given ObjectName are returned. Each GenericConditionObject has the following fields:			
ConditionObject AttributeName	An attribute of the object to be used in a condition.	1..64	Character String representing a valid object attribute. A list of all currently valid attribute names for a supported object can be obtained using the GetGenericObjectAttributes request (see Section 8.25). Also note, a special attribute $COUNT is available. If this attribute name is specified, the query just returns the number of elements.	N/A

Table 312 Generic Query Input Fields (Sheet 3 of 4)

Field Name	Description	Length	Range/Values	Default Value
ConditionObject Logic	Indicates the logical operation that shall be applied using the AttributeName.	enum	equalTo_ notqualTo_ greaterThan_ lesserThan_ in_ like_ greaterThanEqualTo_ lesserThanEqualTo_	N/A
ConditionObject AttributeValue	The value that shall be compared with the provided AttributeName using the provided Logic operation to form this condition.	1..1000	Character String Note: If Logic is “like” then the wildcard symbol % can be present in AttributeValue; e.g.: “1561%” matches all values, which begins with “1561”. If Logic is “in” - then the user can specify comma separated AttibuteValues; e.g. “15615161008, 15615161009" matches values “15615161008” & “15615161009” Also note: If a date value is to be specified, the expected format is CCYY-MM-DDThh:mm[:ss]	N/A
ConditionObject ConditionName	This is an optional parameter. It should be specified only if the ConditionRelationString is specified. See below for details on the use of the ConditionRelationString	1..64	Character String The ConditionName can be any string, “Condition1” for example, but may not contain spaces or special characters.	N/A
ConditionRelationString	This is a string to be provided if a complex condition is required in the query.	1..1000	Character String representing a valid combination of the provided conditions.	N/A

Table 312 Generic Query Input Fields (Sheet 4 of 4)

Table 313 lists the output fields for this method.

Field Name	Description	Length	Range/Values	Default Value
NumRows	The number of rows of data in the current response.	integer		N/A
TableRowDescriptions	This is an array of elements, each of which describes one attribute of the given object and has the following data:			
TableRowDescription RowAttributeName	The name of the attribute.		Character String	N/A

Table 313 Generic Query Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
TableRowDescription RowAttributeType	The type of the attribute.	enum	attrString_ attrDate_ attrNumber_ attrUnsigned_ attrFloat_	N/A
TableRowValues	A two dimensional array of all the requested attribute values of the given ObjectName which matches the given condition(s). It consists of an array of TableRow. Each TableRow has an array of Attribute values. All attribute values irrespective of type are returned as character strings. The position of an Attribute in the TableRowDescriptions matches exactly with the position of the attribute in each TableRow. e.g. if SUBSCRIBER_ID is the 5th attribute in TableRowDescriptions, the value of SUBSCRIBER_ID is present as the 5th element in each TableRow.		array of Character String	N/A

Table 313 Generic Query Input Fields (Sheet 2 of 2)

8.26.1 Generic Query Enhanced

GenericQueryEnhanced shall handle the queries for MLHG, SUBSCRIBER and PREFIX ACCESS CODE entities. It will act as a wrapper and use the existing mechanics of GenericQuery. The **request and response structures will be the same with Generic Query** (which means that xml data will be returned as usual). The internal functionality will add on top of GenericQuery post processing capabilities on the result set of Generic Query.

8.27 Code Process

The Code Process SOAP requests allow provisioning of the Code Processing Table. The user is able to create, delete or query a code processing entry. The modify operation is also allowed after the creation of the code processing with the precondition that the code process entry is not being used (referenced).

In order to create a new code processing entity the user should specify the digit position that he will start to add or delete digits from the called number. Then he has the option to delete a number of digits starting from that position or to add a new digit sequence starting again from the defined digit position. A code processing entity can contain up to four operations (additions or deletions) with the precondition that the maximum number of additions or deletions is two. This means that if a code processing table contains four entries, then two additions and two deletions should occur. During the “digit manipulation”, all the entries for the particular “Code Process” object will apply in the sequence they are provisioned.

8.27.1 Create Code Process

This SOAP request is used to create a Code Processing entry.

Table 314 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
12.00.01.10	N/A	subRC_SUCCESS

Table 314 Create Code Process Return Codes

Table 315 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command. Operator information used for logging.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		required
CodeProcessName	This parameter specifies a user-assigned alphanumeric string that identifies the Code Processing Entry.	1..31	Character string	required

Table 315 Create Code Process Input Fields (Sheet 1 of 3)

Field Name	Description	Length	Range/Values	Default Value
NatureOfAddress	Any valid nature of address value.	enum	NoaIgnored NoaUnused NoaUnassigned NoaUndefined NoaUnkown NoaSubs NoaNatl NoaIntl NoaLrn NoaNw NoaPrfxBased NoaNoDn NoaTest NoaL0 NoaL1 NoaL2 NoaExt NoaPrfxBasedNoTns NoaCallingLoc	NoaUnkown
Retranslate	Indicates whether retranslation will occur. For cases where Retranslate is False, a DestinationName and DestinationType must be provided; if it is True, they should not be provided.	Boolean	false true	false
DestinationName	A name identifying a destination.	0..64	Character string	“”
DestinationType	This attribute is only valid if Retranslate is False. It is allowed for E164 codes.	enum	DestCodeDtIgnored DestCodeDtUnused DestCodeDtNone DestCodeDtDest DestCodeDtSvc DestCodeDtHomeDn DestCodeDtEnum DestCodeDtIntcpt DestCodeDtTod	N/A
NumberOfOperations	The number of code processing entries in the following list.	integer	1..4	N/A
OperationList	An array of CodeOperation elements, each containing the following attributes:			
(CodeOperation) DigitPosition	The index position in the called number string at which to start to insert or delete digits.	integer	0..31 (for DigitsToInsert) OR 1..32 (for DigitsToDelete)	0

Table 315 Create Code Process Input Fields (Sheet 2 of 3)

Field Name	Description	Length	Range/Values	Default Value
(CodeOperation) DigitsToInsert	An alphanumeric string that identifies the digits to be added starting at DigitPosition. Supported characters are A-F, 0-9, # and *. Note that both Insert and Delete are not possible in the same entry.	0..30	Character string	“”
(CodeOperation) DigitsToDelete	The number of digits to delete starting at DigitPosition.	integer	0..31	0

Table 315 Create Code Process Input Fields (Sheet 3 of 3)

8.27.2 Modify Code Process

This SOAP request is used to modify an existing Code Processing entry.

Table 316 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
12.00.01.10	N/A	subRC_SUCCESS

Table 316 Modify Code Process Return Codes

Table 317 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		required
CodeProcessName	This parameter specifies a user-assigned alphanumeric string that identifies the Code Processing Entry.	1..31	Character string	required
NewName	The new name of a CodeProcess	1..31	Character String	optional

Table 317 Modify Code Process Input Fields (Sheet 1 of 3)

Field Name	Description	Length	Range/Values	Default Value
NatureOfAddress	Any valid nature of address value.	enum	NoaIgnored NoaUnused NoaUnassigned NoaUndefined NoaUnkown NoaSubs NoaNatl NoaIntl NoaLrn NoaNw NoaPrfxBased NoaNoDn NoaTest NoaL0 NoaL1 NoaL2 NoaExt NoaPrfxBasedNoTns NoaCallingLoc	N/A
Retranslate	Indicates whether retranslation will occur. For cases where Retranslate is False, a DestinationName and DestinationType must be provided; if it is True, they should not be provided.	Boolean	false true	N/A
DestinationName	A name identifying a destination.	0..64	Character string	N/A
DestinationType	This attribute is only valid if Retranslate is False. It is allowed for E164 codes.	enum	DestCodeDtIgnored DestCodeDtUnused DestCodeDtNone DestCodeDtDest DestCodeDtSvc DestCodeDtHomeDn DestCodeDtEnum DestCodeDtIntcpt DestCodeDtTod	N/A
NumberOfOperations	The number of code processing entries in the following list.	integer	1..4	N/A
OperationList	An array of CodeOperation elements, each containing the following attributes:			
(CodeOperation) DigitPosition	The index position in the called number string at which to start to insert or delete digits.	integer	0..31 (for DigitsToInsert) OR 1..32 (for DigitsToDelete)	N/A

Table 317 Modify Code Process Input Fields (Sheet 2 of 3)

Field Name	Description	Length	Range/Values	Default Value
(CodeOperation) DigitsToInsert	An alphanumeric string that identifies the digits to be added starting at DigitPosition. Supported characters are A-F, 0-9, # and *. Note that both Insert and Delete are not possible in the same entry.	0..30	Character string	N/A
(CodeOperation) DigitsToDelete	The number of digits to delete starting at DigitPosition.	integer	0..31	N/A

Table 317 Modify Code Process Input Fields (Sheet 3 of 3)

8.27.3 Remove Code Process

This SOAP request is used to delete an existing Code Processing entry.

Table 318 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
12.00.01.10	N/A	subRC_SUCCESS

Table 318 Remove Code Process Return Codes

Table 319 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
CodeProcessName	This parameter specifies a user-assigned alphanumeric string that identifies the Code Processing Entry.	1..31	Character string	required

Table 319 Remove Code Process Input Fields

8.27.4 Query Code Process

This SOAP request is used to retrieve and display existing Code Processing entries.

Table 320 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
12.00.01.10	N/A	subRC_SUCCESS

Table 320 Query Code Process Return Codes

Table 321 lists the input fields for this method.

Field Definition	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Specific	Indicates whether a specific object or a list of objects is being requested.	Please refer to the note at the beginning of Chapter 8.		
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			
CodeProcessName	This parameter specifies a user-assigned alphanumeric string that identifies the Code Processing Entry.	1..31	Character string	“”

Table 321 Query Code Process Input Fields

The output fields for this method are the same as the input fields for the Create Code Process SOAP request. Please refer to Section 8.27.1, “Create Code Process” for details.

8.27.5 List Code Process

This SOAP request is used to retrieve a list of existing Code Processing entries.

Table 322 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
12.00.01.10	N/A	subRC_SUCCESS

Table 322 List Code Process Return Codes

Table 323 lists the input fields for this method. The output of the request consists of a list of CodeProcess elements.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Specific	Indicates whether a specific object is being requested, or a list of objects.	Please refer to the note at the beginning of Chapter 8.		
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			
CodeProcessName	This parameter specifies a user-assigned alphanumeric string that identifies the Code Processing Entry.	1..31	Character string	“”

Table 323 List Code Process Input Fields

8.28 ENUM Provisioning

The ENUM (**E**164 (or T**E**lephone) **NU**mber **M**apping) system is used to convert telephone numbers into service specific URIs via DNS lookup. This mechanism allows the routing of calls to other networks through IP-based means instead of through the PSTN.

Through the SOAP interface, the configuration of two ENUM related objects are supported: ENUM Server and Operator ENUM.

8.28.1 Create Enum Server

This SOAP request is used to create an ENUM Server entry. Note that a maximum of six (6) ENUM Servers may be created.

Table 324 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
12.00.01.27	N/A	subRC_SUCCESS

Table 324 Create Enum Server Return Codes

Table 325 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		required
EnumServerName	A unique name identifying an ENUM server.	1..31	Character string	required
PrimaryIpAddress	The primary IP address of the ENUM server.	1..260	Character string	required
SecondaryIpAddress	The secondary IP address of the ENUM server.	1..260	Character string	“”
CacheSize	The size of the ENUM server’s cache, in multiples of 1 MB.	integer	0..10	1

Table 325 Create Enum Server Input Fields

8.28.2 Modify Enum Server

This SOAP request is used to modify an ENUM Server entry.

Table 326 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
12.00.01.27	N/A	subRC_SUCCESS

Table 326 Modify Enum Server Return Codes

Table 327 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		required
EnumServerName	A unique name identifying an ENUM server.	1..31	Character string	required
NewName	The new name of the Enum Server	1..31	Character string	optional
PrimaryIpAddress	The primary IP address of the ENUM server.	1..260	Character string	required
SecondaryIpAddress	The secondary IP address of the ENUM server.	1..260	Character string	“”
CacheSize	The size of the ENUM server’s cache, in multiples of 1 MB.	integer	0..10	1

Table 327 Modify Enum Server Input Fields

8.28.3 Remove Enum Server

This SOAP request is used to delete an existing ENUM Server entry.

Table 328 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
12.00.01.27	N/A	subRC_SUCCESS

Table 328 Remove Enum Server Return Codes

Table 329 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
EnumServerName	A unique name identifying an ENUM server.	1..31	Character string	required

Table 329 Remove Enum Server Input Fields

8.28.4 Query Enum Server

This SOAP request is used to retrieve and display existing ENUM Server entries.

Table 330 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
12.00.01.27	N/A	subRC_SUCCESS

Table 330 Query Enum Server Return Codes

Table 331 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Specific	Indicates whether a specific object or a list of objects is being requested.	Please refer to the note at the beginning of Chapter 8.		
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			

Table 331 Query Enum Server Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
QryEnumServer	A structure containing the following attributes, any of which may be provided as filter criteria for the query.			
EnumServerName	A unique name identifying an ENUM server.	1..31	Character string	“”
PrimaryIpAddress	The primary IP address of the ENUM server.	1..260	Character string	“”
SecondaryIpAddress	The secondary IP address of the ENUM server.	1..260	Character string	“”
CacheSize	The size of the ENUM server’s cache, in multiples of 1 MB.	integer	0..10	0

Table 331 Query Enum Server Input Fields (Sheet 2 of 2)

The output fields for this method are the same as the input fields for the Create Enum Server SOAP request. Please refer to Section 8.28.1 for details.

8.28.5 List Enum Server

This SOAP request is used to retrieve a list of existing ENUM Server entries.

Table 332 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
12.00.01.27	N/A	subRC_SUCCESS

Table 332 List Enum Server Return Codes

Table 333 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Specific	Indicates whether a specific object is being requested, or a list of objects.	Please refer to the note at the beginning of Chapter 8.		
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			
LstEnumServer	A structure containing the following attributes, any of which may be provided as filter criteria for the query.			
EnumServerName	A unique name identifying an ENUM server.	1..31	Character string	“”

Table 333 List Enum Server Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
PrimaryIpAddress	The primary IP address of the ENUM server.	1..260	Character string	“”
SecondaryIpAddress	The secondary IP address of the ENUM server.	1..260	Character string	“”
CacheSize	The size of the ENUM server’s cache, in multiples of 1 MB.	integer	0..10	0

Table 333 List Enum Server Input Fields (Sheet 2 of 2)

The output of the request consists of a list of ENUM Server elements.

8.28.6 Create Operator Enum

This SOAP request is used to create an ENUM Server entry. Note that a maximum of six (6) ENUM Servers may be created.

Table 334 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
12.00.01.27	N/A	subRC_SUCCESS

Table 334 Create Operator Enum Return Codes

Table 335 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		required
OperatorEnumName	A unique name identifying an operator ENUM.	1..15	Character string	required
OperatorEnumTier0ZoneName	A unique name identifying an operator ENUM.	1..64	Character string	required
EnumServerName	The name of an ENUM server.	1..31	Character string	required

Table 335 Create Operator Enum Input Fields

8.28.7 Modify Operator Enum

This SOAP request is used to modify an Operator ENUM entry.

Table 336 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
12.00.01.27	N/A	subRC_SUCCESS

Table 336 Modify Operator Enum Return Codes

Table 337 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		required
OperatorEnumName	A unique name identifying an operator ENUM.	1..15	Character string	required
New OperatorEnumName	The new name of a Operator Enum	1…15	Character string	optional
OperatorEnumTier0ZoneName	A unique name identifying an operator ENUM.	1..64	Character string	required
EnumServerName	The name of an ENUM server.	1..31	Character string	required
OperatorEnumUpdateFlags	A bitmap attribute that indicates which fields are being modified.	integer	Character string	required

Table 337 Modify Operator Enum Input Fields

8.28.8 Remove Operator Enum

This SOAP request is used to delete an existing Operator ENUM entry.

Table 338 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
12.00.01.27	N/A	subRC_SUCCESS

Table 338 Remove Operator Enum Return Codes

Table 339 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
OperatorEnumName	A unique name identifying an operator ENUM.	1..15	Character string	required

Table 339 Remove Operator Enum Input Fields

8.28.9 Query Operator Enum

This SOAP request is used to retrieve and display existing Operator ENUM entries.

Table 340 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
12.00.01.27	N/A	subRC_SUCCESS

Table 340 Query Operator Enum Return Codes

Table 341 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Specific	Indicates whether a specific object or a list of objects is being requested.	Please refer to the note at the beginning of Chapter 8.		
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			
QryOperatorEnum	A structure containing the following attributes, any of which may be provided as filter criteria for the query.			
OperatorEnumName	A unique name identifying an operator ENUM.	1..15	Character string	“”
OperatorEnumTier0ZoneName	A unique name identifying an operator ENUM.	1..64	Character string	“”
EnumServerName	The name of an ENUM server.	1..31	Character string	“”

Table 341 Query Operator Enum Input Fields

The output fields for this method are the same as the input fields for the Create Operator Enum SOAP request. Please refer to Section 8.28.6, “Create Operator Enum” for details.

8.28.10 List Operator Enum

This SOAP request is used to retrieve a list of existing Operator ENUM entries.

Table 342 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
12.00.01.27	N/A	subRC_SUCCESS

Table 342 List Operator Enum Return Codes

Table 343 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
Specific	Indicates whether a specific object is being requested, or a list of objects.	Please refer to the note at the beginning of Chapter 8.		
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			
LstOperatorEnum	A structure containing the following attributes, any of which may be provided as filter criteria for the query.			
OperatorEnumName	A unique name identifying an operator ENUM.	1..15	Character string	“”
OperatorEnumTier0ZoneName	A unique name identifying an operator ENUM.	1..64	Character string	“”
EnumServerName	The name of an ENUM server.	1..31	Character string	“”

Table 343 List Operator Enum Input Fields

The output of the request consists of a list of Operator ENUM elements.

8.29 File System Backup

8.29.1 Create File System Backup

This function is used to request a file system backup. An interface with the Maintenance Manager component of the system is used to perform the operation and a Job ID is returned to the operator so that the status of the backup operation can be queried.

Table 344 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
11.00.12.03	N/A	subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_DSA_FSBACKUP_NODEID_INVALID subRC_APP_DSA_FSBACKUP_CREATE_FAILED subRC_APP_DSA_FSBACKUP_REMOTEIPADDRESS_INVALID subRC_APP_DSA_FSBACKUP_USERLOGIN_INVALID subRC_APP_DSA_FSBACKUP_USERPASSWD_INVALID subRC_APP_DSA_FSBACKUP_REMOTEDIR_INVALID

Table 344 Create File System Backup Return Codes

Table 345 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
NodeId	The host name of the node being addressed	1..16	Character string	required
FSRetrievalMethod	An enumerated value describing how the backup file is to be handled. If FSB_pull is selected, the data will remain on the local host machine for retrieval. If FSB_push is selected, the additional parameters below must be provided.	enum	FSB_pull FSB_push	FSB_pull
RemoteIPAddress	The IP address of the remote location to which the backup file should be sent.	1..24	Character string	“” For FSB_push, this parameter is mandatory; for FSB_pull, this parameter is not allowed.
FSRemoteProtocol	The protocol by which the backup file is to be transferred to the RemoteIPAddress.	enum	FSB_sftp FSB_ftp	FSB_sftp For FSB_pull, this parameter is not allowed.
UserLogin	The login ID of the user to access the remote host.	1..24	Character string	“” For FSB_push, this parameter is mandatory; for FSB_pull, this parameter is not allowed.

Table 345 Create File System Backup Input FIelds (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
UserPassword	The password of the user to access the remote host.	1..24	Character string	"" For FSB_push, if FSRemoteProtocol is FSB_ftp; otherwise this parameter is not allowed.
RemoteDirectory	The directory on the remote host in which the backup file is to be placed.	1..1024	Character string	"" For FSB_push, this parameter is mandatory; for FSB_pull, this parameter is not allowed.

Table 345 Create File System Backup Input FIelds (Sheet 2 of 2)

8.29.2 Cancel File System Backup

This function is used to cancel a previously requested file system backup. An interface with the Maintenance Manager component of the system is used to perform the operation.

Table 346 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
11.00.12.03	N/A	subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_DSA_FSBACKUP_NODEID_INVALID subRC_APP_DSA_FSBACKUP_CANCEL_FAILED

Table 346 Cancel File System Backup Return Codes

Table 347 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, "Field Definitions", on page 67.		
NodeId	The host name of the node being addressed	1..16	Character string	required
JobId	A Job ID identifying the file system backup operation that is to be canceled. Must match the value that was returned from the original request to perform the operation.	integer	any valid integer value	required

Table 347 Cancel File System Backup Input FIelds

8.29.3 Query File System Backup

This function is used to retrieve the status of a previously requested file system backup. An interface with the Maintenance Manager component of the system is used to perform the operation.

Table 348 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
11.00.12.03	N/A	subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_DSA_FSBACKUP_NODEID_INVALID subRC_APP_DSA_FSBACKUP_QUERY_FAILED subRC_APP_DSA_MEMORY_ALLOCATION_ERROR

Table 348 Query File System Backup Return Codes

Table 349 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		required
Paging	The standard paging input structure.	Please refer to Section 5.5, “Paging”, on page 72.		N/A
NodeId	The host name of the node being addressed	1..16	Character string	required
JobId	A Job ID identifying the file system backup operation that is to be queried. Must match the value that was returned from the original request to perform the operation.	integer	any valid integer value	required

Table 349 Query File System Backup Input Fields

Table 350 lists the output fields for this method.

Field Name	Description	Length	Range/Values	Default Value
PaginatedInfo	The standard paging output structure.	For details, refer to Section 5.5, “Paging”, on page 72.		N/A
FileSystemJobList	A list of FileSystemJobEntry elements, each consisting of the following attributes:			
(FileSystemJobEntry) JobId	A Job ID identifying the file system backup operation being queried.	integer	any valid integer value	N/A
(FileSystemJobEntry) ExecPath	Absolute path to the executable (usually a shell script).	system dependent	Character string	N/A

Table 350 Query File System Backup Output Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
(FileSystemJobEntry) Type	Flags that provide information about the job's attributes. For the file system backup job), the flags indicate that the job can be cancelled, can be scheduled, creates a log file and traps on an error.	1..256	Character string	N/A
(FileSystemJobEntry) AvgRunTime	The average time that the job is expected to take. For the file system backup job, this is 3600, and this is intended only as a guide for the operator.	1..20	Character string	N/A
(FileSystemJobEntry) MaxRunTime	The maximum time that the job is expected to take. For the file system backup job, this is 14400, and this is intended only as a guide for the operator.	1..20	Character string	N/A
(FileSystemJobEntry) State	The state of the job -normally one of the following: RUNNING, DONE, CANCELLED or ERROR.	1..256	Character string	N/A
(FileSystemJobEntry) ReportName	The file name for the report file for the job.	system dependent	Character string	N/A
(FileSystemJobEntry) LogName	The file name for the log file for the job. This file should be referred to if the job has a state of ERROR as it will indicate what the error is.	system dependent	Character string	N/A
(FileSystemJobEntry) StartTime	Start time of job.	1..26	Character string	N/A
(FileSystemJobEntry) StopTime	Stop time of job.	1..26	Character string	N/A
(FileSystemJobEntry) MArgs	Arguments passed to executable to execute a job.	1..1024	Character string	N/A

Table 350 Query File System Backup Output Fields (Sheet 2 of 2)

8.30 Get DB Version

8.30.1 Get DB Version Info

Database versioning in the OpenScape Voice system database is a means to validate if a database restore can be performed from a database backup file. The restore is allowed to be executed if the database version of the OpenScape Voice system database and the database version of the database backup file are an exact match. The database version is a combination of the software release ID combined with the database iteration number (i.e ReleaseId.DbId). Each time the database is modified and delivered by means of a software build or a patch, the database version is incremented. This takes place for the entire lifecycle of the release. The software release ID stays constant throughout the entire lifespan of the software release regardless of what build ID is applied to the release. Once a new release is established, a new version ID is associated with that new release. Database iteration IDs are incremented from the starting point of that release.

When a database backup is taken the database version is put into the backup file so that a comparison can be made to the OpenScape Voice system database should a database restore take place. When a database restore is performed the restore script will first check to see that the database version of the backup file and the database version of the OpenScape Voice system are an exact match. If the versions are not an exact match, the restore will be rejected.

Table 351 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
12.00.04.02	N/A	subRC_SUCCESS

Table 351 Get DB Version Info Return Codes

Table 352 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Table 10 in Section 5.4, “Primitive Data Definitions” .		required

Table 352 Get DB Version Info Input Fields

8.31 Endpoint Location

This function is used to provision an Endpoint Location. The endpoint Location expresses the geographical location of an endpoint. Once created it can be associated with an NNI Endpoint.

8.31.1 Create Endpoint Location

Field Name	Description	Mandatory	Length	Range	Default Value
OperatorId	The operator issuing the command.	No	1..40	Character String	N/A
EndpointLocationName	The endpoint location name	No	1….20	Character String	N/A
EndpointLocationCode	The endpoint location	No	1…8	Character String	N/A

Table 353 Create Endpoint Location Input Fields

8.31.2 Modify Endpoint Location

Field Name	Description	Mandatory	Length	Range	Default Value
OperatorId	The operator issuing the command.	No	1..40	Character String	N/A
EndpointLocationName	The endpoint location name	No	1….20	Character String	N/A
EndpointLocationCode	The endpoint location	No	1…8	Character String	N/A

Table 354 Modify Endpoint Location Input Fields

8.31.3 Query Endpoint Location

Field Name	Description	Mandatory	Length	Range	Default Value
OperatorId	The operator issuing the command.	No	1..40	Character String	N/A
EndpointLocationName	The endpoint location name	No	1….20	Character String	N/A
EndpointLocationCode	The endpoint location	No	1…8	Character String	N/A
InterfaceWSDL	Refer to Section 5.1.1, “The Interface WSDL and Interface Release Control”, on page 61				
OperatorID	The operator issuing the command	No	1..40	Character String	N/A

Table 355 Query Endpoint Location Input Fields

Field Name	Description	Mandatory	Length	Range	Default Value
Specific	Indicates whether a specificEndpoint Location is beingrequested, or a list of EndpointLocations.				
PageSize	Paging Standard paging structure forqueries.				
QryEndpointLocationEndpointLocationName	The name of the endpoint location	No	1…20	Character String	N/A
QryEndpointLocationEndpointLocationCode	The code of the endpoint location	No	1…8	Character String	N/A

Table 355 Query Endpoint Location Input Fields

8.31.4 Remove Endpoint Location

Field Name	Description	Mandatory	Length	Range	Default Value
OperatorId	The operator issuing the command.	No	1..40	Character String	N/A
EndpointLocationName	The endpoint location name	No	1….20	Character String	N/A
EndpointLocationCode	The endpoint location	No	1…8	Character String	N/A

Table 356 Remove Endpoint Location Input Fields

8.32 Endpoint Provider

This function is used to provision an Endpoint Provider. The endpoint Provider expresses the telecom service provider for a NNI Endpoint. Note that once a Provider is created it can be associated with an NNI Endpoint only if the NNIType of the specific Endpoint is set to PSTNGateway.

8.32.1 Create Endpoint Provider

Field Name	Description	Length	Range	Default Value
OperatorId	The operator issuing the command.	1..40	Character String	

Table 357 Create Endpoint Provider Input Fields

Field Name	Description	Length	Range	Default Value
EndpointProviderName	The name of the Endpoint Provider	1…12	Character String	N/A

Table 357 Create Endpoint Provider Input Fields

8.32.2 Modify Endpoint Provider

Field Name	Description	Length	Range	Default Value
OperatorId	The operator issuing the command.	1..40	Character String	
OldEndpointProviderName	The old name of the Endpoint Provider	1…12	Character String	N/A
NewEndpointProviderName	The new name of the Endpoint Provider	1…12	Character String	N/A

Table 358 Modify Endpoint Provider Input Fields

8.32.3 Query Endpoint Provider

Field Name	Description	Length	Range	Default Value
InterfaceWSDL	Refer to Section 5.1.1, “The Interface WSDL and Interface Release Control”, on page 61			
OperatorId	The operator issuing the command	1…40	Character String	
Specific	Indicates type of query. If True,the request is for a specificEndpoint Provider; if its is set to False allEndpoint Provider shall be returned.			
Paging	Paging structure for Get Listrequests. For moreinformation, refer to Section 5.5, “Paging”, on page 72			
QryEndpointProvider-EndpointProviderName	The name of the Endpoint Provider	1…12	Character String	N/A

Table 359 Query Endpoint Provider Input Fields

8.32.4 Remove Endpoint Provider

Field Name	Description	Length	Range	Default Value
OperatorId	The operator issuing the command.	1..40	Character String	
EndpointProviderName	The name of the Endpoint Provider	1…12	Character String	N/A

Table 360 Remove Endpoint Provider Input Fields

9 Subscriber

Subscribers are identified by their DN. The subscriber may have many attributes and the following methods are used to manipulate the subscriber and any of their attributes.

9.1 Create Subscriber

Table 361 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
V5.00.20.01	N/A	subRC_SUCCESS subRC_APP_SUB_LEN_INVALID subRC_APP_SUB_DN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_SUB_BILLING_ID_LEN_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_LINE_NAME_LEN_INVALID subRC_APP_BG_EXTENSION_LEN_INVALID subRC_APP_BG_EXTENSION_INVALID subRC_APP_BG_DEPT_NAME_LEN_INVALID subRC_APP_SUB_ACCOUNT_LEN_INVALID subRC_APP_SUB_ACC_USER_INFO_PIN_LEN_INVALID subRC_APP_SUB_ACC_USER_INFO_PIN_INVALID subRC_APP_SUB_ACC_USER_INFO_PUBLIC_PIN_LEN_INVALID subRC_APP_SUB_ACC_USER_INFO_PUBLIC_PIN_INVALID subRC_APP_SUB_USER_INFO_DISPLAY_NAME_OUT_OF_RANGE subRC_APP_UTIL_LANG_LEN_INVALID subRC_APP_UTIL_LANG_NOT_ALLOWED subRC_APP_UTIL_LANG_INVALID subRC_APP_SUB_RETAILER_ID_LEN_INVALID subRC_APP_SUB_STATUS_EXPORT_NOT_ALLOWED subRC_APP_SUB_INTERCEPT_NEW_SERVICE_CONFLICT subRC_APP_SUB_DN_EXIST subRC_DB_SUB_DN_LOOKUP_FAILED subRC_APP_SUB_HOME_DN_DOES_NOT_EXIST subRC_DB_SUB_HOME_DN_LOOKUP_FAILED subRC_APP_SUB_HOME_DN_ALREADY_ALLOCATED subRC_APP_SUB_HOME_DN_REUSE_NOT_ALLOWED subRC_APP_SUB_INTERCEPT_CONNECTION_CONFLICT subRC_APP_SUB_NEW_SERVICE_ID_CONNECTION_CONFLICT subRC_APP_SUB_RCF_CONNECTION_CONFLICT subRC_APP_EP_PROFILE_LEN_INVALID subRC_DB_XDM_EP_LOOKUP_FAILED subRC_APP_EP_SIP_CONNECTION_INVALID subRC_APP_EP_SIP_SIGNALING_IP_ADDRESS_INVALID

Table 361 Create Subscriber Return Codes (Sheet 1 of 4)

WSDL Version	Obsolete in WSDL Version	Return Codes
		subRC_APP_EP_SIP_SIGNALING_PORT_INVALID subRC_APP_EP_SIP_REG_TYPE_INVALID subRC_APP_EP_LATADigit_LEN_INVALID subRC_APP_EP_LATADigit_INVALID subRC_APP_EP_PROFILE_EXIST subRC_APP_EP_REG_TYPE_INVALID subRC_APP_EP_IP_LEN_INVALID subRC_APP_EP_IP_INVALID subRC_APP_EP_PORT_INVALID subRC_APP_EP_STATIC_REG_TYPE_NOT_ALLOWED subRC_APP_EP_SIP_DYNAMIC_IP_CONFLICT subRC_APP_EP_PRE_REG_TYPE_DYNAMIC_CONFLICT subRC_APP_EP_SIP_MAX_SESSIONS_INVALID subRC_DB_EP_SIP_XDM_ALIAS_LOOKUP_FAILED subRC_APP_EP_SIP_XDM_ALIAS_EXIST subRC_APP_EP_SIP_SECURITY_SCHEME_MISSING subRC_APP_EP_SIP_REALM_LEN_INVALID subRC_APP_EP_SIP_USER_NAME_LEN_INVALID subRC_APP_EP_SIP_PASSWORD_LEN_INVALID subRC_APP_EP_SIP_SECURITY_MISSING subRC_APP_EP_PROFILE_LEN_INVALID subRC_APP_SUB_IP_ATTRIBUTES_CONFLICT subRC_DB_XDM_EP_LOOKUP_FAILED subRC_APP_EP_PROFILE_EXIST subRC_APP_EP_ALIAS_LEN_INVALID subRC_DB_COMM_RTM_LOOKUP_FAILED

Table 361 Create Subscriber Return Codes (Sheet 2 of 4)

WSDL Version	Obsolete in WSDL Version	Return Codes
		subRC_APP_EP_VOIP_MISSING subRC_APP_EP_56QOS_INVALID subRC_APP_EP_QOS_QOSPROFILE_CONFLICT subRC_APP_QOS_PROFILE_LEN_INVALID subRC_APP_EP_QOSPRIM_MISSING subRC_APP_EP_PRIM_PKT_CABLE_DYNAMIC_QOS_DATA_MISSING subRC_APP_EP_SEC_PKT_CABLE_DYNAMIC_QOS_DATA_MISSING subRC_APP_EP_QOSTHRD_QOSSEC_CONFLICT subRC_APP_EP_QOSTHRD_QOS4TH_CONFLICT subRC_APP_EP_QOS4TH_QOS5TH_CONFLICT subRC_APP_EP_QOS5TH_QOS6TH_CONFLICT subRC_APP_EP_THRD_PKT_CABLE_DYNAMIC_QOS_DATA_MISSING subRC_APP_EP_4TH_PKT_CABLE_DYNAMIC_QOS_DATA_MISSING subRC_APP_EP_5TH_PKT_CABLE_DYNAMIC_QOS_DATA_MISSING subRC_APP_EP_6TH_PKT_CABLE_DYNAMIC_QOS_DATA_MISSING subRC_APP_SUB_USER_INFO_TIME_ZONE_INVALID subRC_DB_SUB_USER_INFO_TIME_ZONE_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_SUB_OUT_OF_RANGE subRC_APP_BG_CPU_ID_INVALID subRC_APP_BG_CPU_ID_BG_ID_MISSING subRC_DB_BG_NUMBER_CPU_LOOKUP_FAILED subRC_APP_BG_CPU_SUB_OUT_OF_RANGE subRC_APP_SUB_BG_NAME_MISSING subRC_APP_BG_EXTENSION_BG_NAME_MISSING subRC_APP_BG_ATT_NUM_BG_NAME_MISSING subRC_DB_BG_ATT_NUM_INSERTION_FAILED subRC_APP_BG_DEPT_BG_NAME_MISSING subRC_DB_BG_DEPT_DOES_NOT_EXIST subRC_DB_SUB_PIC_LOOKUP_FAILED subRC_APP_EP_T38_ECH_CAN_CONFLICT subRC_APP_QOS_PROFILE_SUB_CONFLICT subRC_APP_EP_QOSPRIM_PACK_PER_INVALID subRC_APP_EP_QOSCLASS_QOSVALUE_INVALID subRC_APP_EP_QOSSEC_PACK_PER_INVALID subRC_APP_EP_QOSSEC_TOS_INVALID subRC_APP_EP_QOSTHRD_PACK_PER_INVALID subRC_APP_EP_QOSTHRD_TOS_INVALID

Table 361 Create Subscriber Return Codes (Sheet 3 of 4)

WSDL Version	Obsolete in WSDL Version	Return Codes
		subRC_APP_EP_QOS4TH_PACK_PER_INVALID subRC_APP_EP_QOS4TH_TOS_INVALID subRC_APP_EP_QOS5TH_PACK_PER_INVALID subRC_APP_EP_QOS5TH_TOS_INVALID subRC_APP_EP_QOS6TH_PACK_PER_INVALID subRC_APP_EP_QOS6TH_TOS_INVALID subRC_APP_EP_CODEC_PACK_PER_INVALID subRC_APP_FP_NAME_LEN_INVALID subRC_APP_SUB_NUM_SUBSCRIBER_OUT_OF_RANGE subRC_DB_SUB_INSERTION_FAILED subRC_APP_SUB_NUM_SUBSCRIBER_OUT_OF_RANGE subRC_APP_EP_NUMBER_SUB_OUT_OF_RANGE subRC_DB_EP_INSERTION_FAILED subRC_APP_EP_SIP_SECURITY_NUMBER_OUT_OF_RANGE subRC_DB_EP_SIP_SEC_INSERTION_FAILED subRC_DB_COMM_RTM_CREATION_FAILED subRC_DB_COMM_RTM_LOOKUP_FAILED subRC_DB_SUB_XDM_UNASSIGN_FAILED subRC_DB_SUB_XDM_ASSIGN_FAILED subRC_APP_SUB_FP_PROPAGATION_FAILED subRC_APP_SUB_CREATION_CPU_FAILED subRC_APP_SUB_KEYSET_NUMBER_OUT_OF_RANGE subRC_DB_SUB_KEYSET_INSERTION_FAILED subRC_APP_SUB_USER_INFO_UNICODE_DISPLAY_NAME_OUT_OF_RANGE subRC_APP_UNICODE_BG_LINE_NAME_LEN_INVALID subRC_APP_NOT_ALLOWED_SETTING_CPU_TO_MOBILEDN subRC_APP_NOT_ALLOWED_SETTING_KEYSET_TO_MOBILEDN subRC_APP_SUB_MOBL_RING_DUR_INVALID subRC_APP_SUB_MAIN_RING_DUR_INVALID subRC_APP_SUB_CELL_RING_DUR_INVALID subRC_APP_FEATURES_AND_OSMO_CONFLICT subRC_APP_FEATURE_PROFILE_OSMO_CONFLICT subRC_APP_SUB_OSMO_FP_CONFLICT subRC_APP_SUB_OSMO_CPG_MEMBER_CONFLICT subRC_APP_SUB_OSMO_COM_MEMBER_CONFLICT subRC_APP_SUB_OSMO_EA_MEMBER_CONFLICT subRC_APP_SUB_OSMO_KEYSET_CONFLICT subRC_APP_OSMO_TLS_CONFLICT subRC_APP_OSMO_EXTERNAL_DN_FLAG_CONFLICT subRC_APP_EP_INSECURE_REFERRED_BY_HEADER_OSMO_CONFLICT subRC_APP_SUB_ROUTETOCELL_CELLDEVICENUMBER_CONFLICT

Table 361 Create Subscriber Return Codes (Sheet 4 of 4)

Table 362 lists and describes the CreateSubscriber fields. Included are the Length, Range/Values, and Default Values for each field.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
ServiceId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
AccountMgtInfo				
Account ID	Used only for auditing and aiding the telco in identifying the subscriber. This is an optional parameter and there is no default This field is not used by OpenScape Voice call processing. The only way to use this field is via SOAP export and offline processing.	1..20	Character string.	“”
Billing ID	Used only for auditing and aiding the telco in identifying the subscriber. This is an optional parameter and there is no default. This field is not used by OpenScape Voice call processing. The only way to use this field is via SOAP export and offline processing.	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		“”
Retailer ID	Used as an auditing field for the telco to help the customer identify the owner of the subscriber. The retailer is optional and there is no default as part of the subscriber definition. The value is not required to match data in the Retailer/ Carrier table. This field is not used by OpenScape Voice call processing. The only way to use this field is via SOAP export and offline processing.	1..16	Character string.	“”
COS (Class of Service)	Outgoing calls are routed using the Class of Service. During routing, if a subscriber belongs to a class of service, those routes are given preference; otherwise, the switch’s default class of service (unassigned) is used. If the subscriber has no Class of Service assigned, the subscriber can access default routes only. Subscribers with COS specified will access COS routes first (if defined for the call that they make), if none are available, the default COS routes will be used.	1..15	Character string representing any existing ClassOfService.[1]	“”

Table 362 Create Subscriber Field Definitions (Sheet 1 of 14)

Field Name	Description	Length	Range/Values	Default Value
RateArea	The subscriber can belong to a specific Rate Area. This Rate Area defines what constitutes a local call verses a long distance call, as well as the routing for outbound calls. Outgoing calls are routed using the Rate Area. During routing, if a subscriber belongs to a Rate Area, those routes are given preference; otherwise, the switch's default rate area (unassigned) is used. If the subscriber has no Rate Area assigned, the subscriber can access default routes only.	1..15	Character string representing any existing RateArea.[2]	""
TestLineFlag	Indicates whether the line is a test line. Outbound calls do not behave any differently from non-test lines except that the test line indication is set in the calling party category (SS7 message). CDRs will be generated but there should be no charges generated for the call. Supported in Japan only.	Boolean	True, False	False
ExternalDN Flag	Indicates that the subscriber DN is a Direct Inward Dialing (DID) number. The subscriber can receive direct incoming calls. If the External DN Flag is set to False, the BGL has a private (non-external) number. This means the number cannot be dialed from outside of the BG and is not displayed as a calling party on calls directed outside the BG. Note that this flag is also used by the MLHG service when an MLHG member originates a call. If True, the terminal's own data is used as the calling party; otherwise, the Pilot DN's data is used.	Boolean	True, False Must be true when subscriber is OSMO.	True

Table 362 Create Subscriber Field Definitions (Sheet 2 of 14)

Field Name	Description	Length	Range/Values	Default Value
CountryCode Len	To support international dialing and multiple countries being served by the same OpenScape Voice system, the country code will be part of the subscriber DN. The country code is required to be created as part of the office code in the Home Dn. The subscriber country code length must match the office code country code length. The format and valid range are controlled by the RTP parameters specified by the SrxDigitLen.parm file. The Country Code Length is automatically retrieved from the HomeDn and assigned to the subscriber during subscriber creation. Therefore, it is no longer an input parameter. It cannot be modified and will be displayed as part of the subscriber information.	integer	1..4	N/A
CallingLoc	Indicates the subscriber's location. Used for routing emergency calls to the answering point that serves the Calling Location.	1..15	Any existing Calling Location.[3]	""
FeatureProfile Id, FeatureProfile Name	Identifies the subscriber's Feature Profile, a defined set of features that is inherited by the subscriber by virtue of association with the Feature Profile.	For further details, see definition of FeatureProfileName in Section 5.4, "Primitive Data Definitions", on page 66.		""
FeatureProfile Id, BGName	The name of the Business Group with which the Feature Profile is associated. A key field for identification of a Feature Profile. It must be the same as the subscriber's BGName	For further details, see definition of BGName in Section 5.4, "Primitive Data Definitions", on page 66.		""
NumberPlanName	The name of the Private Numbering Plan to associate with this BGL. If none is provided, this BGL uses the BG's numbering plan. For residential subscribers, only the system default numbering plan applies.	1..31	Character string.	""
ExternalCallerID		Char array	15 (1…15)	N/A
UseMainPilotDN		boolean	True False	false

Table 362 Create Subscriber Field Definitions (Sheet 3 of 14)

Field Name	Description	Length	Range/Values	Default Value
ImpactLevel	The security zone of a subscriber. WSDL V7.00.15.01	enumeration	ImpactLevelZero, ImpactLevelOne, ImpactLevelTwo, ImpactLevelThree, ImpactLevelFour,> ImpactLevelFive, ImpactLevelSix, ImpactLevelSeven, ImpactLevelEight ImpactLevelNine ImpactLevelUnclassified	
SubscriberBGInfo				
BGName	Business Group Name: optional for this request. For further details, see Section 5.4, “Primitive Data Definitions”, on page 66.	0..30	Up to 30 characters, including apostrophes, commas, and other special characters.	required
BGLineName	Business Group Line Name is the subscriber member name.	0..20	Character string	“”
UnicodeBGLineName	A BG Line name for use in markets such as China where multi-byte characters are allowed.	1..24	Wide character string.	“”
BGAttendant Number	A business group attendant number flag indicates the DN is used for the Attendant Number function of the Business Group.	Boolean	True, False	False
CpuId	A Call Pickup group Id to which this subscriber (BGL) belongs. There can be only one CPU group ID per subscriber. The CfCPU service must be provisioned either by explicit assignment or Feature Profile inheritance in order for the CPU service to be effective. A value of 0 indicates that the subscriber is not part of any CPU.	integer	1..10000	0 (no CPU Id)
CpuName			Character string	
Extension	This field indicates the Extension number of the subscriber (BGL). Optional; used for intra BG calls display only.	1..7	Numeric string	“”
BGDeptName	The department name associated with the subscriber.	1..20	Character string	“”

Table 362 Create Subscriber Field Definitions (Sheet 4 of 14)

Field Name	Description	Length	Range/Values	Default Value
AccountUserInfo				
DisplayName	Used for in-switch and VPN dialing to present the calling party's name. Used in installations where the SCP is unavailable or the switch is configured to use this field instead of the SCP. This field may be up to 20 characters in length; note, however, that in some networks, ISUP supports only 15 characters. In these environments, only the first 15 characters of the field will be sent. SIP phones may populate a generic name field that contains an alphabetical name to be associated with the number. If populated, that field will be used instead of this field. *Optional.*	1..20	Character string.	""
UnicodeDisplayName	A display name for use in markets such as China where multi-byte characters are allowed.	1..24	Wide character string.	""
TimeZone	Specifies the subscriber's time zone location and determines how the subscriber's time-dependent services work. In general, most subscribers will be in the switch's local time zone, but SIP & H323 phones can be anywhere and the switch could border a time zone edge and serve multiple time zones.		Any valid time zone.[4]	"Local"
Pin	An optional field specifying the subscriber's Personal Identification Number. This code is available for use by various service features where a PIN has meaning, unless that feature has its own PIN. The subscriber PIN may be up to 10 characters long. Service feature pins may be up to 7 characters long.	1..10	Character string	""
Pin2	Optional field specifying additional subscriber's Personal Identification Numbers. All PINs (Pin-Pin5) are unique to a subscriber. PINs Pin2-Pin5 are available for use only during COSS service activation/deactivation. During COSS service activation and deactivation, the COSS service will validate the dialed PIN from all the PINs configured for that subscriber. Pin2-Pin5 may be up to 10 characters long. The valid characters for PINs are digits 0 to 9	1..10	Character string	""
Pin3		1..10	Character string	""
Pin4		1..10	Character string	""
Pin5		1..10	Character string	""

Table 362 Create Subscriber Field Definitions (Sheet 5 of 14)

Field Name	Description	Length	Range/Values	Default Value
PublicPin	A Public PIN provided by the subscriber that can be used by users who dial in. For example, the ACR service requests a PIN before allowing the call to terminate to a subscriber. The Public PIN associated with the destination subscriber is expected to be input. Note: added in WSDL version 10.00.08.01.	1..10	Character string	“”
Language	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			“System Default”

Table 362 Create Subscriber Field Definitions (Sheet 6 of 14)

Field Name	Description	Length	Range/Values	Default Value
Status				
Subscriber Status	The administrative state of the subscriber. For more information on SubscriberStatus, refer to UpdateSubscriberStatus request in Section 9.7, "Update Subscriber Status", on page 346.	enum	Active, Blocked	Active
Intercept Announcement	The intercept announcement to which calling parties are routed if this subscriber's number is disconnected or changed.	enum	None, Disconnect, NumberChange	N/A
NewServiceId	This field is used when a subscriber's ServiceId has changed. It may optionally be played during the NumberChange announcement.	See definition of ServiceId in Section 5.4, "Primitive Data Definitions", on page 66.		""
ConnectionInfo	The subscriber may be a "profile-only" subscriber (no real phone) or may have a physical phone. If the subscriber has a physical phone, it must be defined during the create. Only one type of phone connection may be specified. The following parameters are prefixed with the substructure that specifies the type of connection for which they apply. These substructures are called IPPhone			
(IPPhone) Name	The name that the phone will register with. An internal switch alias is created of type GUI_ALIAS_TYPE_SIPURL. Note that if the SIP device is configured to register by number, this Name must exactly match the DN of the subscriber in order for registration to be successful. Otherwise, this Name should match the name that the SIP device registers with.	1..15	Character string	required
(IPPhone) RegType	The Registration type. **Static**: – For Static registration, the Signaling port and IP of the phone must be sent. IP phones connected through a Gateway may be pre-registered. Note that Static registration is not supported for Keyset subscribers. **Dynamic** – The signaling Port may be sent. If it is not, it will default to 5060. If the IPConnectionType is set to TLS the port will default to 5061. The request will be rejected if the IP Address is sent. Note that if the registration type is changed from Static to Dynamic via the UpdateConnectionInfo request, the IP Address, if any was present, will be cleared, and the Port, if not sent, will be set back to the default (5060).	enum	Static, Dynamic	required

Table 362 Create Subscriber Field Definitions (Sheet 7 of 14)

Field Name	Description	Length	Range/Values	Default Value
(IPPhone) IPAddr	The IP address or domain name of the phone. Allowed to be set if the SIP phone is defined to be statically registered.	1..256	standard IP address convention: 0.0.0.0..255.255.255.255	N/A
(IPPhone) Port	The static port number used for registration. Meant for RegType=Static.		standard port number convention: 1..9999	5060 If the IPConnectionType is set to TLS, the port will default to 5061.
(IPPhone) LataDigits	Identifies the LATA to which the subscriber belongs. (The switch may serve subscribers across multiple latas.) This information is used during SCP 800 number translations to an SCP.	0..3	Numeric string	“”
(IPPhone) Register	When the RegType is Static, the phone may be pre-registered and not required to register when it is powered on. The setting of this parameter is blocked if RegType is Dynamic.	Boolean	True, False	False
(IPPhone) TimeToLive	The time left from when the phone registers to when it must reregister before being forcibly unregistered by the switch. Note: Read-only parameter, no longer supported in CreateSubscriber request.		A timestamp (seconds since 1970) when the phone will expire.	N/A
(IPPhone) MaxSessions	Indicates the maximum number of conversation sessions that should be allowed simultaneously - note these are messaging sessions, not voice (audio) sessions.		1..10000 (expected to be limited to 10 for an individual SIP subscriber, but the actual upper limit is the value defined in the RTP parameter “Srx/Sip/max_number_of_sessions”	3
(IPPhone, SIPSecurity[5]) Realm	The domain of the SIP Phone. If the switch is configured to require SIP Authentication (Srx/Sip/AuthEnabled), this field is required	1..128	Character string	“”
(IPPhone, SIPSecurity) UserName	The SIP User Name. This value may be different from the Directory Number.	1..64	Character string	“”
(IPPhone, SIPSecurity) Password	The password that is used in conjunction with the UserName.	1..20	Character string	“”

Table 362 Create Subscriber Field Definitions (Sheet 8 of 14)

Field Name	Description	Length	Range/Values	Default Value
(IPPhone, SIPSecurity) Scheme	The type of security scheme that is being implemented. In this release, only digest_authentication form of security is implemented. Specifying no security removes any previously configured security information for this subscriber.	enum	digest_authentication, no_security	no_ security
(IPPhone, SIPSecurity) SrtpSupport	SRTP Support	Enum	SrtpUnassigned SrtpAutomatic SrtpEnabled SrtpDisabled	SrtpDisable d
(IPPhone, SIPSecurity) SdesSrtpSupport	"Best-Effort SRTP with SDES" attribute to determine whether to include the SRTP media in the SDP offer or not.	Enum	SrtpUnassigned SrtpAutomatic SrtpEnabled SrtpDisabled	SrtpDisable d
(IPPhone) IPConnection Type	The type of connection used.	enum	UDP, TCP, TLS	UDP
(IPPhone) ContactInfoList	Statically registered phones may have multiple lists.			N/A
(IPPhone) TrustedSub	Indicates that the subscriber is behind a trusted host. If a subscriber is provisioned as subscriber located behind a trusted host, a P-Asserted-Identity header will be sent always by the signaling manager. Note that this attribute only applies when the OpenScape Voice system is configured with the Customer Type of Communication Server or Real Time Server.	Boolean	True, False	False
(IPPhone Attributes) MediaReDirectionAll owed	Indicates whether this SIP subscriber is configured as a static IMS subscriber. It is used for static subscribers who don't register to indicate that the route header URL providing a route to the IMS is stored in the LooseRoutingURI attribute.	Boolean	True, False	False
(IPPhone Attributes) StaticIMSSub	Indicates whether this SIP subscriber is configured as a static IMS subscriber.	Boolean	True, False	False
(IPPhone) AssociatedSipEndp ointName	The name of the SIP Endpoint that can handle calls between registered SIP subscribers in case the OpenScape Voice system is not accessible for call control. Displays a list of all the SIP Endpoints that don't have a subscriber profile. Also known as the SIP Survivable Proxy.	1..15	Character string	""

Table 362 Create Subscriber Field Definitions (Sheet 9 of 14)

Field Name	Description	Length	Range/Values	Default Value
(IPPhone) LooseRoutingURI	A string of up to 256 characters representing the contact URI of the subscriber. Valid only for a static subscriber when application type is 2 (voice application server for FMC) and the StaticIMSSub attribute is true. By default, if not provided on creation, the text populated in the RTP parameter Srx/Sip/default_s_cscf_URI is used.	1..256	Character string	The text defined in the RTP parameter: Srx/Sip/default_s_cscf_URI
(IPPhone) AcceptBillingNumber	A charge number received in a XSiemens-CDR header field of a SIP INVITE or REFER request (for blind call transfer) will be included as the “ANI/Billing number” (field13.) of the CDR for the call.	Boolean	false true	false
(IPPhone) SendInsecureReferredByHeader	When a Referred-By header field is included in the initial request sent to the transfer target (for a blind transfer), the Referred-By header (including any generic parameters) will be passed transparently from the transferor to the transfer target.	Boolean	false true Must be true when subscriber is OSMO.	true
(IPPhone Attribute) OverrideIRMCodecRestrictions	Force selection of Codecs by overriding IRM restrictions	Boolean	false true	false
(IPPhone Attributes) TransferHandoff	If this attribute is set, calls can be transferred to the Genesys	Boolean	false, true	false
Video Call Allowed	If the attribute is set the Video Call are allowed	Boolean	false, true	false
(IPPhone Attributes) AnatSetting	Indicates whether Proxy/SBC ANAT settings for calls to subscribers are used. The attribute is applicable only to proxy endpoints (‘SIP proxy’ attribute set).	Boolean	false, true	false

Table 362 Create Subscriber Field Definitions (Sheet 10 of 14)

Field Name	Description	Length	Range/Values	Default Value
(IPPhone) AnatSupport	Indicates whether the subscriber's endpoint supports ANAT. AnatOn indicates that the endpoint supports ANAT. AnatOff means that the endpoint does not support ANAT. AnatAuto means whether the endpoint supports ANAT shall be determined by the SIP registrar during registration of the endpoint. AnatUnassigned indicates that the default value should be assigned,	Enum	AnatUnassigned AnatAuto AnatOn, AnatOff	If the registration of the subscriber is static: AnatOn If the registration of the subscriber is dynamic: AnatAuto (For the subscriber's registration type please refer to IPPhone RegType field)
Reserved2	The purpose for this is to be prepared for future urgent requests requiring a new endpoint attribute and where new functionality via 8K (e)patch is delivered but there is no time to deliver the corresponding Assistant changes at the same time. The intention is to use these reserve attributes only in exceptional cases where the regular process of adding endpoint attributes can not be applied due to time constraints. Usage of these reserve attributes must be explicitly approved by development management before proceeding.	Boolean	True, False	False
Reserved3		Boolean	True, False	False
Reserved4		Boolean	True, False	False
(IPPhone Attributes) SendUriInTelephone SubscriberFormat	If the attribute is set , the TEL URI is sent in telephone subscriber (global number or local number) format (including phone context for local numbers)	Boolean	false, true	false
(Attribute) SendInternationalNumberInGNF	New name for the SendUriInTelephone SubscriberFormat attribute	Boolean	True, False	false

Table 362 Create Subscriber Field Definitions (Sheet 11 of 14)

Field Name	Description	Length	Range/Values	Default Value
(IPPhone Attributes) DisableLongCallAu	When enabled, the long call duration timer is disabled.	Boolean	True, False	false
ExternalCallerID		Char array	15 (1…15)	N/A
UseMainPilotDNInternalCalls	Optional. Set 1 means that main pilot DN is used for internal calls. When set 0 means that main pilot DN is not used for internal calls.	boolean	True False	false
UseMainPilotDNExternalCalls	Optional. Set 1 means that main pilot DN is used for external calls. When set 0 means that main pilot DN is not used for external calls.	boolean	True False	false
MobileFlag	Enables OSMO capability	Boolean	False true	false
MainDeviceDN	The associated Deskphone	Please refer to Section 5.4, “Primitive Data Definitions”, on page 80.		
CellDeviceNumber	Cell phone number of an OSMO. V7.00.20.01 and forward	string	[1-29] ‘+’ before number is required	NA
RouteToCell	Option to route the call to the mobile phone (cell device number) V7.01.01.01	boolean	[false,true] Must not be true when CellDeviceNumber is not defined	NA
MobileDeviceRingDuration	Ring duration for the Mobile subscriber	integer	5,10,15 up to V7.00.23.01) 1-25 from V7.00.24.01	NA
MainDeviceRingDuration	Ring duration for the Deskphone	integer	5,10,15 up to V7.00.23.01) 1-25 from V7.00.24.01	NA
CellDeviceRingDuration	Ring duration for the cellular phone	integer	5,10,15 up to V7.00.23.01) 1-25 from V7.00.24.01	NA
SubscriberLicenseType	Indicates the current dynamic license status usage for this subscriber. Normally corresponds to the registered status of the subscriber - if the subscriber is registered, it’s status is DynamicSubscriberLicense; otherwise it is UndefinedSubscriberLicense.	enum	UndefinedSubscriberLicense DynamicSubscriberLicense	N/A
VOIP - Not for OpenScape Voice.				
Bearer_Speech	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			True
Bearer_3_1K	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False
Bearer_56K	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False

Table 362 Create Subscriber Field Definitions (Sheet 12 of 14)

Field Name	Description	Length	Range/Values	Default Value
Bearer_64K	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False
FaxMode	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			G711
Echo Cancellation	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			True
Silence Suppression	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False
FaxStatistics	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False
OutbandDTMF Signaling	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False
Suppress OutbandDTMFWith G711	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False
PIC1	Obsolete Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			“”
PIC2	Obsolete Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			“”
PIC3	Obsolete Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			“”
PIC4	Obsolete Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			“”
KeysetInfo				
KeysetUse	Describes whether a DN is to be used for Keyset Operation and, if so, in what way. For descriptions of the possible values, refer to Section 4.13, “Keyset Information”, on page 55 for more information.	enum	KeysetPrimaryLine KeysetPhantomLine KeysetNone	Keyset None
ListOfFeatures				
A list of service features to be applied to the created subscriber/BGL. For more information, refer to SOAP request *UpdateSubscriberFeatures* in Section 9.14, “Update Subscriber Features”. Note that if a FeatureProfile is provided as part of the same CreateSubscriber request, validation of feature interactions between the features in the FeatureProfile and the features included in this list is not currently supported. If a subscriber is to have both an associated FeatureProfile and directly assigned features, it is recommended that the subscriber be created first with the FeatureProfile and any additional features added later using the *UpdateSubscriberFeatures* SOAP request.				

Table 362 Create Subscriber Field Definitions (Sheet 13 of 14)

Field Name	Description	Length	Range/Values	Default Value
Notes: [1]Defined ClassesOfService can be retrieved using the *GetCOSList* interface method (see Section 7.6, "Get COS List"). They can be defined on the switch using either the OpenScape Voice Assistant or CLI. [2]The available list of Rate Areas can be retrieved using the *GetRateAreaList* interface method (see Section 7.7, "Get Rate Area List"). They can be defined using either the OpenScape Voice Assistant or CLI. [3]The available list of Calling Locations can be retrieved using the *GetCallingLocationList* interface method (see Section 7.9, "Get Calling Location List"). [4]For a list of valid time zones, refer to Appendix A, "Time Zones". [5]The following structure contains fields for when SIP Authentication has been enabled on the platform. These fields are required when the RTPParm "Srx/Sip/Authentication-Enabled" = RtpTrue				

Table 362 Create Subscriber Field Definitions (Sheet 14 of 14)

9.2 Get Subscriber Info

The *GetSubscriberInfo* SOAP request is used to retrieve the subscriber profile. Note that the most current request is GetSubInfoByRel; the others are retained for backward compatibility with older clients, but are not expected to be used by new clients.

A ServiceId is required to query a subscriber's information, and this is normally the subscriber's DN (Directory Number). However, it is also possible to provide an Alias in the ServiceId field, if the subscriber has been provisioned with one or more alphanumeric aliases. If the input ServiceId is numeric only, the method will assume it is a DN, while if it is in valid Alias URI format (see definition of the AliasURI parameter in the Create Subscriber request), the Alias will be used to query the subscriber. The subscriber DN will be returned, and the Alias information will be in the ConnectionInfo substructure.

- **GetSubscriberInfo** — returns only the tags common to older clients.
- **GetSubscriberByOption** — supports fixed clients that require the latest interface.
- **GetSubInfoByRel** — The request contains release version and a list of the attributes for which data is to be returned. *If no attribute descriptors are provided, ALL attributes are returned.*

Note: It is possible to perform a *GetSubscriberInfo* request with a disconnected DN, but there is no data stored in the subscriber profile table for these DNs, so the only information returned is the status information that can be obtained from the Home DN table. These are the following attributes: SubscriberStatus (always **Blocked**), InterceptAnnouncement, and NewServiceId, if any.

Table 363 lists the data returned in response to the *GetSubscriberInfo* requests:

Field Name	Description	Length	Range/Values	Default Value
ServiceId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
OfficeCodeInfo				
OfficeCode	Numeric identifier of a specific office code.		Must be a digit sequence of maximum length 9. N.America: 6 digits Japan: 5 digits Korea: 4 or 5 digits Europe: 1 to9 digits	
CountryCodeLength	Length of the country code prefix in the office code.	Int	0 - 4 in all markets	0
AreaCodeLength	Length of the area code prefix in the office code.	Int	N.America: 3 Japan: 1-5 Korea: 1-2 China: 2-3 Europe: 0-5	
Attributes	Defines attributes of the office code. Enumerated type.	5-digit string		
AccountMgmtInfo				
Account ID	Used only for auditing and aiding the telco in identifying the subscriber. This is an optional parameter and there is no default	1..20	Character string.	N/A
Billing ID	A DN to which charges are assigned in the generated CDRs. It is optional and if not present, the subscriber's DN is used.	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		N/A
Retailer Id	Used as an auditing field for the telco to help the customer identify the owner of the subscriber. The retailer is optional and there is no default as part of the subscriber definition. The value is not required to match data in the Retailer/ Carrier table.	1..16	Character string.	N/A
COS (Class of Service)	Outgoing calls are routed using the Class of Service. During routing, if a subscriber belongs to a class of service, those routes are given preference; otherwise, the switch’s default class of service (unassigned) is used. If the subscriber has no Class of Service assigned, the subscriber can access default routes only. Subscribers with COS specified will access COS routes first (if defined for the call that they make), if none are available, the default COS routes will be used.	1..15	Character string representing any existing ClassOfService.	N/A

Table 363 Get Subscriber Info Data Returned (Sheet 1 of 11)

Field Name	Description	Length	Range/Values	Default Value
RateArea	The subscriber can belong to a specific Rate Area. This Rate Area defines what constitutes a local call verses a long distance call, as well as the routing for outbound calls. Outgoing calls are routed using the Rate Area. During routing, if a subscriber belongs to a Rate Area, those routes are given preference; otherwise, the switch's default rate area (unassigned) is used. If the subscriber has no Rate Area assigned, the subscriber can access default routes only.	1..15	Character string representing any existing RateArea.	N/A
TestLineFlag	Indicates whether the line is a test line. Outbound calls do not behave any differently from non-test lines except that the test line indication is set in the calling party category (SS7 message). CDRs will be generated but there should be no charges generated for the call. Supported in Japan only.	Boolean	True, False	N/A
ExternalDN Flag	Indicates that the subscriber DN is a Direct Inward Dialing (DID) number. The subscriber can receive direct incoming calls. If the External DN Flag is set to False, the BGL has a private (non-external) number. This means the number cannot be dialed from outside of the BG and is not displayed as a calling party on calls directed outside the BG. Note that this flag is also used by the MLHG service when an MLHG member originates a call. If True, the terminal's own data is used as the calling party; otherwise, the Pilot DN's data is used.	Boolean	True, False	N/A

Table 363 Get Subscriber Info Data Returned (Sheet 2 of 11)

Field Name	Description	Length	Range/Values	Default Value
CountryCode Len	To support international dialing and multiple countries being served by the same hiQ, the country code will be part of the subscriber DN. The country code is required to be created as part of the office code in the HomeDN. The subscriber country code length must match the office code country code length. The format and valid range are controlled by the RTP parameters specified by the SrxDigitLen.parm file. The Country Code Length is automatically retrieved from the HomeDN and assigned to the subscriber during subscriber creation. Therefore, it is no longer an input parameter. It cannot be modified and will be displayed as part of the subscriber information.	integer	1..4	N/A
CallingLoc	Indicates the subscriber's location. Used for routing emergency calls to the answering point that serves the Calling Location.	1..15	Any existing Calling Location.	N/A
FeatureProfile Id, FeatureProfile Name	Identifies the subscriber's Feature Profile, a defined set of features that is inherited by the subscriber by virtue of association with the Feature Profile.	For further details, see definition of FeatureProfileName in Section 5.4, "Primitive Data Definitions", on page 66.		N/A
FeatureProfile Id, BGName	The name of the Business Group with which the Feature Profile is associated. A key field for identification of a Feature Profile. It must be the same as the subscriber's BGName	For further details, see definition of BGName in Section 5.4, "Primitive Data Definitions", on page 66.		N/A
NumberPlanName	The name of the Private Numbering Plan to associate with this BGL. If none is provided, this BGL uses the BG's numbering plan. For residential subscribers, only the system default numbering plan applies.	1..31	Character string.	N/A
SubscriberBGInfo				
BGName	Business Group Name: optional for this request. For further details, see Section 5.4, "Primitive Data Definitions", on page 66.	0..30	Up 30 characters, including apostrophes, commas, and other special characters.	required
BGLineName	Business Group Line Name is the subscriber member name.	0..20	Character string	N/A
UnicodeBGLineNam e	A BG Line name for use in markets such as China where multi-byte characters are allowed.	1..24	Wide character string.	N/A

Table 363 Get Subscriber Info Data Returned (Sheet 3 of 11)

Field Name	Description	Length	Range/Values	Default Value
BGAttendant Number	A business group attendant number flag indicates the DN is used for the Attendant Number function of the Business Group.	Boolean	True, False	N/A
CpuId	A Call Pickup group Id to which this subscriber (BGL) belongs. There can be only one CPU group ID per subscriber. The CfCPU service must be provisioned either by explicit assignment or Feature Profile inheritance in order for the CPU service to be effective. A value of 0 indicates that the subscriber is not part of any CPU.	integer	1..10000	N/A
CpuName			Character string	
Extension	This field indicates the Extension number of the subscriber (BGL). Optional; used for intra BG calls display only.	1..7	Numeric string	N/A
BGDeptName	The department name associated with the subscriber.	1..20	Character string	N/A
BGBranchOfficeName	The name of the Branch Office the subscriber queried for belong to.	1-30	Character String	""

Table 363 Get Subscriber Info Data Returned (Sheet 4 of 11)

Field Name	Description	Length	Range/Values	Default Value
AccountUserInfo				
DisplayName	Used for in-switch and VPN dialing to present the calling party's name. Used in installations where the SCP is unavailable or the switch is configured to use this field instead of the SCP. This field may be up to 20 characters in length; note, however, that in some networks, ISUP supports only 15 characters. In these environments, only the first 15 characters of the field will be sent. SIP phones may populate a generic name field that contains an alphabetical name to be associated with the number. If populated, that field will be used instead of this field. *Optional.*	1..20	Character string.	N/A
UnicodeDisplayNam e	A display name for use in markets such as China where multi-byte characters are allowed.	1..24	Wide character string.	N/A
TimeZone	Specifies the subscriber's time zone location and determines how the subscriber's time-dependent services work. In general, most subscribers will be in the switch's local time zone, but SIP & H323 phones can be anywhere and the switch could border a time zone edge and serve multiple time zones.		Any valid time zone.	N/A
Pin	An optional field specifying the subscriber's Personal Identification Number. This code is available for use by various service features where a PIN has meaning, unless that feature has its own PIN. The subscriber PIN may be up to 10 characters long. Service feature pins may be up to 7 characters long.	1..10	Character string	N/A
PublicPin	A Public PIN provided by the subscriber that can be used by users who dial in. For example, certain services, such as EACR may request a PIN before allowing the call to terminate to a subscriber. The Public PIN associated with the destination subscriber is expected to be input. Note: added in WSDL version 10.00.08.01.	1..10	Character string	N/A
Language	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.			N/A

Table 363 Get Subscriber Info Data Returned (Sheet 5 of 11)

Field Name	Description	Length	Range/Values	Default Value
Status				
Subscriber Status	The administrative state of the subscriber.	enum	Active, Blocked	N/A
Intercept Announcement	The intercept announcement to which calling parties are routed if this subscriber's number is disconnected or changed.	enum	None, Disconnect, NumberChange	N/A
NewServiceId	This field is used when a subscriber's ServiceId has changed. It may optionally be played during the NumberChange announcement.	See definition of ServiceId in Section 5.4, "Primitive Data Definitions", on page 66.		N/A
ConnectionInfo	The subscriber may be a "profile-only" subscriber (no real phone) or may have a physical phone. If the subscriber has a physical phone, it must be defined during the create. Only one type of phone connection may be specified. The following parameters are prefixed with the substructure that specifies the type of connection for which they apply. These substructures are called IPPhone.			
(IPPhone) Name	The name that the phone will register with. An internal switch alias is created of type GUI_ALIAS_TYPE_SIPURL. Note that if the SIP device is configured to register by number, this Name must exactly match the DN of the subscriber in order for registration to be successful. Otherwise, this Name should match the name that the SIP device registers with.	1..15	Character string	N/A
(IPPhone) RegType	The Registration type. **Static**: – For Static registration, the Signaling port and IP of the phone must be sent. IP phones connected through a Gateway may be pre-registered. Note that Static registration is not supported for Keyset subscribers. **Dynamic** – The signaling Port may be sent. If it is not, it will default to 5060. The request will be rejected if the IP Address is sent. Note that if the registration type is changed from Static to Dynamic via the UpdateConnectionInfo request, the IP Address, if any was present, will be cleared, and the Port, if not sent, will be set back to the default (5060).	enum	Static, Dynamic	N/A
(IPPhone) IPAddr	The IP address or domain name of the phone. Allowed to be set if the SIP phone is defined to be statically registered.	1..256	standard IP address convention: 0.0.0.0..255.255.255.255	N/A
(IPPhone) Port	The static port number used for registration. Meant for RegType=Static.		standard port number convention: 1..9999	N/A

Table 363 Get Subscriber Info Data Returned (Sheet 6 of 11)

Field Name	Description	Length	Range/Values	Default Value
(IPPhone) LataDigits	Identifies the LATA to which the subscriber belongs. (The switch may serve subscribers across multiple latas.) This information is used during SCP 800 number translations to an SCP.	0..3	Numeric string	N/A
(IPPhone) Register	When the RegType is Static, the phone may be pre-registered and not required to register when it is powered on. The setting of this parameter is blocked if RegType is Dynamic.	Boolean	True, False	N/A
(IPPhone) TimeToLive	The time left from when the phone registers to when it must reregister before being forcibly unregistered by the switch. Note: Read-only parameter, no longer supported in CreateSubscriber request.		A timestamp (seconds since 1970) when the phone will expire.	N/A
(IPPhone) MaxSessions	Indicates the maximum number of sessions that should be allowed simultaneously - note that these are messaging sessions, not voice (audio) sessions.		1..10000 (expected to be limited to 10 for an individual SIP subscriber, but the actual upper limit is the value defined in the RTP parameter *Srx/Sip/ max_number_of_sessions*	N/A
(IPPhone, SIPSecurity[5]) Realm	The domain of the SIP Phone. If the switch is configured to require SIP Authentication (*Srx/Sip/AuthEnabled*), this field is required	1..128	Character string	N/A
(IPPhone, SIPSecurity) UserName	The SIP User Name. This value may be different from the Directory Number.	1..64	Character string	N/A
(IPPhone, SIPSecurity) Password	The password that is used in conjunction with the UserName.	1..20	Character string	N/A
(IPPhone, SIPSecurity) Scheme	The type of security scheme that is being implemented. In this release, only digest_authentication form of security is implemented. Specifying no security removes any previously configured security information for this subscriber.	enum	digest_authentication, no_security	N/A
SrtpSupport	SRTP Support	Enum	SrtpUnassigned SrtpAutomatic SrtpEnabled SrtpDisabled	SrtpDisabled
SdesSrtpSupport	“Best-Effort SRTP with SDES” attribute to determine whether to include the SRTP media in the SDP offer or not.	Enum	SrtpUnassigned SrtpAutomatic SrtpEnabled SrtpDisabled	SrtpDisabled

Table 363 Get Subscriber Info Data Returned (Sheet 7 of 11)

Field Name	Description	Length	Range/Values	Default Value
(IPPhone) IPConnection Type	The type of connection used.	enum	UDP, TCP, TLS	N/A
(IPPhone) ContactInfoList	Statically registered phones may have multiple lists.			N/A
(IPPhone Attributes) TrustedSub	Indicates that the subscriber is behind a trusted host. If a subscriber is provisioned as subscriber located behind a trusted host, a P-Asserted-Identity header will be sent always by the signaling manager. Note that this attribute only applies when the hiQ is configured with the Customer Type of Communication Server or Real Time Server.	Boolean	True, False	N/A
(IPPhone Attributes) GSMSub	Indicates that the subscriber is a GSM Mobile IP Centrex subscriber. If both the Calling and Called Party are GSM subscribers, a configurable prefix will be inserted before sending an INVITE to the trusted host by the signaling manager to avoid the Called party route to the SIP network a second time. A P-Asserted Identity header will also be sent by the signaling manager for a GSM Subscriber. Note that this attribute only applies when the hiQ is configured with the Customer Type of Communication Server or Real Time Server.	Boolean	True, False	N/A
(IPPhone Attributes) MediaReDirectionAllowed	Indicates whether or not media redirection is allowed for this SIP subscriber.	Boolean	True, False	N/A
(IPPhone Attributes) StaticIMSSub	Indicates whether this SIP subscriber is configured as a static IMS subscriber. It is used for static subscribers who don't register to indicate that the route header URL providing a route to the IMS is stored in the LooseRoutingURI attribute.	Boolean	True, False	N/A
(IPPhone Attributes) AORPerContact	Allows the support of single and multiple registration subscribers on the same switch. If set to False, the switch will only allow one AoR per endpoint, either statically or dynamically registered. If set to True, the switch will accept one AoR per DN and per each of the two aliases, store them in the contact and use them for terminating calls.	Boolean	True, False	N/A

Table 363 Get Subscriber Info Data Returned (Sheet 8 of 11)

Field Name	Description	Length	Range/Values	Default Value
(IPPhone) AssociatedSipEndpointName	The name of the SIP Endpoint that can handle calls between registered SIP subscribers in case the OpenScape Voice system is not accessible for call control. Also known as the SIP Survivable Proxy.	1..15	Character string	N/A
(IPPhone) LooseRoutingURI	A string of up to 256 characters representing the contact URI of the subscriber. Valid only for a static subscriber when application type is 2 (voice application server for FMC) and the StaticIMSSub attribute is true. By default, if not provided on creation, the text populated in the RTP parameter Srx/Sip/default_s_cscf_URI is used.	1..256	Character string	N/A
(IPPhone) AliasList	An array of up to 2 AliasEntry elements. Note that in order for a subscriber to make use of these entries, the parameter AORPerContact must be True. Each element contains the following attribute:			
(AliasEntry) AliasURI	A string of representing the alias URI of the subscriber. Please refer to the definition of AliasURI in the Create Subscriber request for information about valid strings for this field.	1..60	Character string	N/A
(IPPhone) SIPUserInterfaceType	If provisioned, indicates that the subscriber is a trusted virtual subscriber located in the TDM network. Corresponds internally to the connection type SIPanalogUserIF. This attribute cannot be modified after creation.	enum	UIAnalog	N/A
(IPPhone) AcceptBillingNumber	A charge number received in a XSiemens-CDR header field of a SIP INVITE or REFER request (for blind call transfer) will be included as the “ANI/Billing number” (field13.) of the CDR for the call.	Boolean	false true	false
(IPPhone) SendInsecureReferredByHeader	When a Referred-By header field is included in the initial request sent to the transfer target (for a blind transfer), the Referred-By header (including any generic parameters) will be passed transparently from the transferor to the transfer target.	Boolean	false true	true
(IPPhone Attribute) OverrideIRMCodecRestrictions	Force selection of Codecs by overriding IRM restrictions.	Boolean	false true	false

Table 363 Get Subscriber Info Data Returned (Sheet 9 of 11)

Field Name	Description	Length	Range/Values	Default Value
(IPPhone Attributes) TransferHandoff	If this attribute is set, calls can be transferred to the Genesys.	Boolean	false, true	false
(IPPhone Attributes) AnatSetting	Indicates whether Proxy/SBC ANAT settings for calls to subscribers are used. The attribute is applicable only to proxy endpoints ('SIP proxy' attribute set).	Boolean	false, true	false
VideoCallAllowed	If the attribute is set the Video Call are allowed	Boolean	false, true	false
SubscriberLicenseType	Indicates the current license status usage for this subscriber. Normally corresponds to the registered status of the subscriber - if the subscriber is registered, it's status is DynamicSubscriberLicense. If the subscriber is registered as an OSMO device it's status is OSMOSubscriberLicense. Otherwise it Is UndefinedSubscriberLicense.	Enum	UndefinedSubscriberLicense DynamicSubscriberLicense OSMOSubscriberLicense	N/A
(IPPhone) VIPType	Refers to the VIPType of the subscriber.	enum	VIPNo, VIPYes	VIPNo
(IPPhone) AnatSupport	Indicates whether the subscriber's endpoint supports ANAT. AnatOn indicates that the endpoint supports ANAT. AnatOff means that the endpoint does not support ANAT. AnatAuto means whether the endpoint supports ANAT shall be determined by the SIP registrar during registration of the endpoint. AnatUnassigned indicates that the default value should be assigned,	Enum	AnatUnassigned AnatAuto AnatOn, AnatOff	If the registration of the subscriber is static: AnatOn If the registration of the subscriber is dynamic: AnatAuto (For the subscriber's registration type please refer to IPPhone RegType field)
SubscriberLicenseType	Indicates the current dynamic license status usage for this subscriber. Normally corresponds to the registered status of the subscriber - if the subscriber is registered, it's status is DynamicSubscriberLicense; otherwise it is UndefinedSubscriberLicense.	enum	UndefinedSubscriberLicense DynamicSubscriberLicense	N/A

Table 363 Get Subscriber Info Data Returned (Sheet 10 of 11)

Field Name	Description	Length	Range/Values	Default Value
VOIP Not for OpenScape Voice.				
Bearer_Speech	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
Bearer_3_1K	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
Bearer_56K	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
Bearer_64K	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
FaxMode	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
Echo Cancellation	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
Silence Suppression	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
FaxStatistics	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
OutbandDTMF Signaling	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
Suppress OutbandDTMFWith G711	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
PIC1	Please refer to the definition of InterExchange.PIC1 in Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
PIC2	Please refer to the definition of InterExchange.PIC2 in Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
PIC3	Please refer to the definition of InterExchange.PIC3 in Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
PIC4	Please refer to the definition of InterExchange.PIC4 in Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
KeysetInfo				
KeysetUse	Describes whether a DN is to be used for Keyset Operation and, if so, in what way.	enum	KeysetPrimaryLine KeysetPhantomLine KeysetNone	N/A
ListOfFeatures				
A list of service features to be applied to the created subscriber/BGL. For more information, refer to SOAP request *UpdateSubscriberFeatures* in Section 9.14 on page 9-361.				

Table 363 Get Subscriber Info Data Returned (Sheet 11 of 11)

Attention: This request is superseded by the Get Sub Info By Rel request in Section 9.2.2, "Get Sub Info By Rel" below, but the information about input and output data included here still applies.

Table 364 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
1.0.1	N/A	subRC_SUCCESS subRC_APP_SUB_LEN_INVALID subRC_APP_SUB_DN_DOES_NOT_EXIST subRC_DB_SUB_DN_LOOKUP_FAILED subRC_DB_SUB_OFFICE_CODE_LOOKUP_FAILED subRC_DB_SUB_HOME_DN_LOOKUP_FAILED subRC_DB_XDM_EP_LOOKUP_FAILED subRC_DB_SUB_TEEN_LOOKUP_FAILED subRC_DB_SUB_FEATURE_LOOKUP_FAILED subRC_APP_SUB_PROFILE_TYPE_INVALID subRC_DB_BG_LOOKUP_FAILED subRC_DB_BG_DEPT_LOOKUP_FAILED subRC_DB_UTIL_LANG_LOOKUP_FAILED subRC_DB_EP_H323_XDM_ALIAS_LOOKUP_FAILED subRC_DB_EP_SIP_XDM_ALIAS_LOOKUP_FAILED subRC_DB_EP_SIP_SECURITY_DATA_LOOKUP_FAILED subRC_DB_EP_XDM_CONTACT_LOOKUP_FAILED subRC_DB_SUB_KEYSET_LOOKUP_FAILED subRC_DB_MLHG_FEAT_DATA_LOOKUP_FAILED subRC_DB_MLHG_PILOT_DN_LOOKUP_FAILED subRC_DB_MLHG_TERM_LOOKUP_FAILED

Table 364 Get Subscriber Info Return Codes

Table 365 lists the field descriptions for this method.

Field Name	Description	Length	Range/Value	Default Value
ServiceId	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.			N/A
GetSubData Options Get Subscriber DataOption List Get Subscriber DataOption	Select a list of all data information needed from the subscriber profile.	List of enums	IncludeAccountMgtInfo_, IncludeAccountUserInfo_, IncludeStatus_, IncludeConnectionInfo_, IncludeVOIP_, IncludeQualityOfService_, IncludeSubscriberAuditInfo_, IncludeListOfFeatures_, IncludeBGInfo_, IncludeKSInfo_, IncludeMlhgInfo_, IncludeContactInfo_,	All values included

Table 365 Get Subscriber Info Input Fields

9.2.1 Get Subscriber Info By Option

This method retrieves selected portions of the subscriber profile.

Attention: This request is superseded by the Get Sub Info By Rel request in Section 9.2.2, “Get Sub Info By Rel” below.

Table 366 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
1.0.1	N/A	Same as GetSubscriberInfo above.

Table 366 Get Subscriber Info By Option Return Codes

Table 367 lists the field descriptions for this method.

Field Name	Description	Length	Range	Default Value
ServiceId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
GetSubData Options. Get Subscriber DataOption List. Get Subscriber DataOption	Select a list of all data information needed from the subscriber profile.	List of enums	Same as GetSubscriberInfo above.	

Table 367 Get Subscriber Info By Option Input Fields

9.2.2 Get Sub Info By Rel

This method retrieves all or part of a subscriber’s data based on an interface release version.

Table 368 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
1.0.1	N/A	Same as GetSubscriberInfo above.

Table 368 Get Sub Info By Rel Return Codes

Table 369 lists the field descriptions for this method.

Field Name	Description	Length	Range	Default Value
Interface Release	The release version being used by the requesting SOAP client.	Please refer to the definition of InterfaceRelease in Section 5.1.1, “The Interface WSDL and Interface Release Control”, on page 61.		Latest
ServiceId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
GetSubData Options. Get Subscriber DataOption List. Get Subscriber DataOption	Select a list of all data information needed from the subscriber profile.	List of enums	Same as GetSubscriberInfo above.	
InterfaceWSDL	The WSDL version being used by the requesting SOAP client.	For more information, refer to Section 5.1.1, “The Interface WSDL and Interface Release Control”, on page 61.		

Table 369 Get Sub Info By Rel Field Descriptions

9.2.3 Get Subscriber RG Request

The GetSubscriberRGRequest SOAP request is used to retrieve part of the subscriber data. A ServiceId is required to query a subscriber's information, and this is normally the subscriber's DN (Directory Number). Additionally, it is possible to send the operator ID, which is a non-mandatory field.

Attention: It is also possible to retrieve a disconnected subscriber through this new API. A message is displayed indicating that the requested subscriber is disconnected. The other data related to the subscriber status, InterceptAnnouncement, and the new service ID cannot be displayed through this API. To display all data, refer to the previous GetSubscriber Info SOAP method (Section 9.2, “Get Subscriber Info”, on page 320)

Table 370 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
1.0.1	N/A	Same as GetSubscriberInfo above.

Table 370 Get Subscriber RG Request Return Codes

Table 371 lists the field descriptions for this method.

Field Name	Description	Length	Range	Default Value
ServiceId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
User Name	The SIP User Name. This value may be different from the Directory Number.	1..64	Character String	N/A
Password	The password that is used in conjunction with the UserName.	1..20	Character String	N/A
Realm	The domain of the SIP Phone.	1..128	Character String	N/A
Rate Area	The subscriber can belong to a specific Rate Area. This Rate Area defines what constitutes a local call vs. a long distance call, as well as the routing for outbound calls. Outgoing calls are routed using the Rate Area. During routing, if a subscriber belongs to a Rate Area, those routes are given preference; otherwise, the system's default rate area (unassigned) is used. If the subscriber has no Rate Area assigned, the subscriber can access default routes only.	1..15	Character string representing any existing RateArea.	N/A
BGName	Business Group Name: optional for this request. For further details, Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.	0..30	Up 30 characters, including apostrophes, commas, and other special characters.	required
MlhgOwner	Indicates if the subscriber is a MLHG owner.	Boolean	True, False	false
PrimaryPilotServiceId	The service id of the PrimaryPilot	1..15	Character String	N/A
MlhgList	Returns the list of MLHG the requested subscriber belongs to	1 …2048 list members		N/A
SubMediaReDirectionAllowed	Indicates whether this SIP subscriber is configured as a static IMS subscriber. It is used for static subscribers who do not register to indicate that the route header URL providing a route to the IMS is stored in the LooseRoutingURI attribute.	Boolean	True, False	

Table 371 Get Subscriber RG Request Field Descriptions

9.3 Get Subscriber List

The GetSubscriberList method is used to retrieve the list of subscribers on the switch. A starting subscriber can be provided or part of a subscriber ID containing the wildcard character (%), and a maximum count to be retrieved.

The default behavior is that subscribers are returned in numerical order from lowest DN to highest. Sorting can also be performed on several other characteristics that can be filled in the GetSubListSortElement list of the request. A number of filter criteria are possible so that a particular subset of existing subscribers can be retrieved. When a specific filter is implemented as a list, multiple filter value selection is available. Wildcarding is also available for string attributes and range/min/max is implemented for the CpuID attribute. All conditions specified are AND-ed together.

The highest supported and default maximum number of subscribers to retrieve is 1000. However, the total number of existing subscribers that meet the given filter criteria is returned in the TotalSubsFound field so that if there are more than 1000 existing subscribers, the client application can send multiple requests, each one with starting subscriber greater than the last one returned and thus obtain all existing subscribers.

The response list will contain disconnected DNs only if the InterceptAnnouncement filter criterion is provided. Disconnected DNs are not stored in the subscriber profile table, but will have their InterceptAnnouncement set to either Disconnect or NumberChange in the Home DN table.

Table 372 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
1.0.1	N/A	subRC_SUCCESS subRC_APP_SUB_SQL_FILTER_OUT_OF_RANGE subRC_APP_SUB_GET_LIST_OUT_OF_RANGE subRC_DB_UTIL_COS_LOOKUP_FAILED subRC_APP_UTIL_COS_DOES_NOT_EXIST subRC_DB_UTIL_RATEAREA_LOOKUP_FAILED subRC_APP_UTIL_RATEAREA_DOES_NOT_EXIST subRC_DB_UTIL_CALLINGLOC_LOOKUP_FAILED subRC_APP_UTIL_CALLINGLOC_DOES_NOT_EXIST subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DEPT_BG_ID_MISSING subRC_DB_BG_DEPT_LOOKUP_FAILED subRC_APP_BG_CPU_ID_BG_ID_MISSING subRC_APP_SUB_PROVISIONING_DATE_INVALID subRC_APP_QOS_PROFILE_LEN_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_QOS_PROFILE_DOES_NOT_EXIST subRC_APP_FP_NAME_LEN_INVALID subRC_APP_FP_DOES_NOT_EXIST subRC_APP_SUB_FEAT_FILTER_NOT_ALLOWED subRC_APP_SUB_FEAT_HD_DUPLICATE subRC_DB_SUB_FEATURE_LOOKUP_FAILED subRC_SYS_SUB_LIST_ALLOC_ERR subRC_DB_SUB_DN_LOOKUP_FAILED subRC_SYS_SUB_FEAT_HD_ALLOC_ERR subRC_SYS_SUB_LIST_ALLOC_ERR subRC_APP_BG_CPU_ID_FILTER_COMBINATION_INVALID

Table 372 Get Subscriber List Return Codes

Table 373 lists the field descriptions for this method.

Attention: The Default Value column is N/A for most parameters, since these input parameters are filter criteria. If a parameter is not provided, the default behavior is that there is no filtering performed based on that particular parameter.

Field Name	Description	Length	Range	Default Value
InterfaceWSDL	The WSDL version of the request. This indicates that only data that was introduced in the specified WSDL version or later shall be returned in response to this request. For further details, please refer to Section 5.1, “Interface (WSDL) Versions”, on page 61.			
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			N/A
ServiceId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
MaxSubscribers	The maximum number of subscribers to be returned in the response list.	integer	N/A	1000
COS	Class of Service. During routing, if a subscriber belongs to a class of service, those routes are given preference; otherwise, the switch’s default class of service (unassigned) is used. If the subscriber has no Class of Service assigned, the subscriber can access default routes only.	Refer to CreateSubscriber in Section 9.1, “Create Subscriber”, on page 303.		N/A
RateArea	Defines what constitutes a local call verses a long distance call, as well as the routing for outbound calls.	Refer to CreateSubscriber in Section 9.1, “Create Subscriber”, on page 303.		N/A
AccountId	Used only for auditing and aiding the telco in identifying the subscriber.	Refer to CreateSubscriber in Section 9.1, “Create Subscriber”, on page 303.		N/A
RetailerId	Used as an auditing field to help the customer identify the owner of the subscriber.	Refer to CreateSubscriber in Section 9.1, “Create Subscriber”, on page 303.		N/A
BillingId	The BGL/DN to which charges are assigned in the CDRs.	Refer to CreateSubscriber in Section 9.1, “Create Subscriber”, on page 303.		N/A
TestLineFlag	Indicates whether the line is a test line. Outbound calls do no behave any differently from non-test lines except that the test line indication is set in the calling party category (SS7 message). Japan only.	Refer to CreateSubscriber in Section 9.1, “Create Subscriber”, on page 303.		N/A
Subscriber Status	Indicates whether a subscriber's calls are active or blocked.	Refer to CreateSubscriber in Section 9.1, “Create Subscriber”, on page 303.		N/A

Table 373 Get Subscriber List Field Descriptions (Sheet 1 of 5)

Field Name	Description	Length	Range	Default Value
Intercept Announcement	Refer to CreateSubscriber in Section 9.1, “Create Subscriber”, on page 303.			N/A
PIC1	Obsolete Refer to CreateSubscriber in Section 9.1, “Create Subscriber”, on page 303.			N/A
PIC2	Obsolete Refer to CreateSubscriber in Section 9.1, “Create Subscriber”, on page 303.			N/A
PIC3	Obsolete Refer to CreateSubscriber in Section 9.1, “Create Subscriber”, on page 303.			N/A
PIC4	Obsolete Refer to CreateSubscriber in Section 9.1, “Create Subscriber”, on page 303.			N/A
BGName	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
BranchOfficeName	The name of the Branch Office the subscribers queried for belong to	1-30	Character String	""
CpuId	Indicates the call pickup group to which this subscriber (BGL) belongs. There can be only one CPU group ID per subscriber. The CfCPU service must be provisioned either by explicit assignment or Feature Profile inheritance in order for the CPU service to be effective. A value of 0 indicates that the subscriber is not part of any CPU.	Int	1..10000	N/A
MinCpuId	If filled in alone, the function will return all subscribers with CPU group ID above the one specified. In conjuction with MaxCpuid a range is provided. Cannot be used concurrently with CpuId	Int	1..10000	N/A
MaxCpuId	If filled in alone, the function will return all subscribers with CPU group ID below the one specified. In conjuction with MaxCpuid a range is provided. Cannot be used concurrently with CpuId	Int	1..10000	N/A

Table 373 Get Subscriber List Field Descriptions (Sheet 2 of 5)

Field Name	Description	Length	Range	Default Value
ProvDateOp Last	Records returned will be entries logged after this specified date.	xsd__date Time	Any valid data/time string.	N/A
CallingLoc	A location, such as an area code.	1 – 15		N/A
BGDeptName	The Business Group Department name on which to filter.	Please refer to the definition of BgDeptName in Table 497		
BgDeptNameList	Used to provide multiple filter value selection for BGDeptName. Cannot be used concurrently with BGDeptName			
isMlhgPilot	If true, specifies that only MLHG Pilot DNs should be returned.	Boolean	True, False	N/A
isMlhgMember	If true, specifies that only MLHG Members should be returned.	Boolean	True, False	N/A
SILMATagIsEnabl ed	Filters by Subscribers that have SILMA Tag enabled (WSDL V7.00.09.01)	Boolean	True, False	N/A
NumberPlanName	The name of the Private Numbering Plan on which to filter. Only BGLs that have been explicitly assigned this Number Plan will be returned.	1..31	Character string.	“”
NumberPlanName List	Used to provide multiple filter value selection for NumberPlanName. Cannot be used concurrently with BGDeptName		List of Char strings	
ConnectionType	The type of connection used.	enum	UDP, TCP, TLS	UDP
ResrcMgmtType	Specifies the Resource Management Type of the QoS Profile. The value PCMM indicates that this QoS Profile is used for Packet Cable MultiMedia. NOTE - PCMM does not apply to OpenScape Voice.	enum	ResourceMgmtType_None ResourceMgmtType_PCMM	ResourceMgmtT ype_None
GetSubListSortList GetSubListSortTy pe	Used to provide multiple filter value selection for Sorting elements. Please refer to Table 374 for a list of possible values.	enum		N/A
GetSubListSortList SortOrder	Provides selection of sorting order	enum	SortOrderAscending, SortOrderDescending	SortOrderAscend ing
DataOptionList GetSubscriberList DataOption	Provides selection of extra data fields to be included in the result structure. Please refer to Table 375 for a list of possible values.	enum		N/A

Table 373 Get Subscriber List Field Descriptions (Sheet 3 of 5)

Field Name	Description	Length	Range	Default Value
GetSubscriberList Da taOptionList GetSubscriberList Da taOption	Select a list of additional data from the subscriber profile database table.	enum	SubListDataInclExtCallerID_ SubListDataInclBOName_, SubListDataInclMobileDN_ SubListDataInclAssocEpIP_ SubListDataInclTransportType_ SubListDataInclSipSecurity SubListDataInclIsMainDevice_ SubListDataInclCpuName SubListDataInclMlhgList	N/A
Assigned Features	A list of features on which to filter. Only subscribers that have the given features assigned will be returned.		Following features supported: BgLC CCBS(V7.00.01.02) CCNR (V7.00.01.02)	N/A
Filter by Assigned Features	A list of features on which to filter. Only subscribers that have the given features assigned will be returned.		List of pointers to all possible features on which filtering can be performed.	N/A
Excluded BGNames	A list of Business Group Names that will be filtered out from the results of GetSubscriberList			N/A
BGLineName	The subscriber member name. A string containing the wildcard (%) character can also be provided. The function will then return matching entries.	0..20	Character string	“”
UnicodeBGLineNa me	A BG Line name for use in markets such as China where multi-byte characters are allowed. A string containing the wildcard (%) character can also be provided. The function will then return matching entries.	1..24	Wide character string.	“”
Extension	This field indicates the Extension number of the subscriber (BGL). It is an optional field and being used for intra BG calls display only.	7	Numeric string	“”
DisplayName	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. A string containing the wildcard (%) character can also be provided. The function will then return matching entries.	1..20	Character string	N/A

Table 373 Get Subscriber List Field Descriptions (Sheet 4 of 5)

Field Name	Description	Length	Range	Default Value
UnicodeDisplayName	A display name for use in markets such as China where multi-byte characters are allowed. A string containing the wildcard (%) character can also be provided. The function will then return matching entries.	1..24	Character string.	“”
FeatureProfileId, FeatureProfileName	Identifies the subscriber’s Feature Profile, a defined set of features that is inherited by the subscriber by virtue of association with the Feature Profile.	For further details, see definition of FeatureProfileName in Section 5.4, “Primitive Data Definitions”, on page 66.		N/A
FeatureProfileId, BGName	The name of the Business Group with which the Feature Profile is associated. A key field for identification of a Feature Profile. It must be the same as the subscriber’s BGName	For further details, see definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		N/A
KeysetUse	Describes whether a DN is to be used for Keyset Operation and, if so, in what way. For descriptions of the possible values, refer to Section 4.13, “Keyset Information”, on page 55.	enum	KeysetPrimaryLine KeysetPhantomLine KeysetNone	N/A
KeysetUseList	Used to provide multiple filter value selection for KeysetUse.	enum		N/A

Table 373 Get Subscriber List Field Descriptions (Sheet 5 of 5)

Table 374 lists the *input* fields for GetSubListSortType.

Field Name	Description	Length	Range	Default Value
GetSubListSortList.GetSubListSortType	Select a list of all sorting elements needed from the subscriber profile.	enum	SubListSortServiceID_ SubListSortBGLineName_ SubListSortUnicodeBGLineName_ SubListSortCpuid_ SubListDataInclCpuId_ SubListSortBGDeptName_ SubListSortDisplayName_ SubListSortUnicodeDisplayName_ SubListSortFeatureProfileId_ SubListSortNumberPlanName_	N/A

Table 374 GetSubListSortType Input Fields

Table 375 lists the *input* fields for GetSubscriberListDataOption.

Field Name	Description	Length	Range	Default Value
GetSubscriberListDataOptionList GetSubscriberListDataOption	Select a list of additional data from the subscriber profile database table.	enum	SubListDataInclBGName_ SubListDataInclBGLineName_ SubListDataInclUnicodeBGLineName_ SubListDataInclCpuId_ SubListDataInclCpuName_ SubListDataInclExtension_ SubListDataInclBGDeptName_ SubListDataInclDisplayName_ SubListDataInclUnicodeDisplayName_ SubListDataInclFeatureProfileId_ SubListDataInclNumberPlanName_ SubListDataInclMlhgOwner_ SubListDataInclMlhgMember_ SubListDataInclKeysetUse_ SubListDataInclOfficeCode_ SubListDataInclMobileDn_] SubListDataInclAssocEpIP_ SubListDataInclTransportType_ SubListDataInclSipSecurity_ SubListDataInclIsMainDevice SubListDataInclCpuName SubListDataInclMlhgList	N/A

Table 375 GetSubscriberListDataOption Input Fields

9.4 Get TombStone Subscriber List

This SOAP request provides the ability to display a list of subscribers deleted after a specified date and/or the list of subscribers that has had a specific service unassigned (currently only CSTA is supported) after a specified date.

The tombstone table is a table used to log deleted subscriber entries and unassigned service entries for a DN.

Table 376 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.03.03		subRC_APP_UTIL_TOMB_ENTRY_TIME_INVALID subRC_APP_UTIL_TOMB_SQL_FILTER_OUT_OF_RANGE subRC_APP_UTIL_TOMB_ENTRY_TYPE_FILTER_NOT_ALLOWED subRC_SYS_UTIL_TOMB_ALLOC_ERR subRC_INTERNAL_ERROR subRC_SUCCESS

Table 376 Get TombStone Subscriber List Return Codes

Table 377 lists the *input* fields for this method.

Attention: The Default Value column is N/A for most parameters, since these input parameters are filter criteria. If a parameter is not provided, the default behavior is that there is no filtering performed based on that particular parameter.

Field Name	Description	Length	Range	Default Value
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			
EntryTimeStamp	Records returned will be entries logged after this specified date.	xsd__date Time	Any valid data/time string.	N/A
EntryType	Filtering condition. If the EntryType is not specified, all records are returned. Else the user can choose to request DeletedSubscribers or ServiceRemoved.		TombStoneDeleteSubscriber, TombStoneDeleteService	N/A
Features	Valid only when the EntryType = TombStone_DeleteService. Filtering Condition. Only unassigned features that match any of the input features will be returned.	N/A	List of pointers to all possible features on which filtering can be performed.	N/A
BGName	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required

Table 377 Get TombStone Subscriber List Input Fields

9.5 Get Keyset Primary List

Given a particular DN, returns a list of all subscribers that are Keyset Primary Lines having the given DN as a Secondary Line Appearance. A maximum of 40 subscribers are returned, since that is the maximum number of devices on which a particular DN can appear as a Secondary Line Appearance.

Table 378 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.04.08	N/A	subRC_SYS_SUB_KEYSET_ALLOC_ERR subRC_APP_SUB_KEYSET_LINE_INVALID subRC_APP_SUB_KEYSET_LINE_MISSING subRC_DB_SUB_KEYSET_LINE_LOOKUP_FAILED subRC_SUCCESS

Table 378 Get Keyset Primary List Return Codes

Table 379 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
Line Appearance	The DN of a keyset line appearance for which the list of primary keyset lines is being requested.	See definition of ServiceId in Section 5.4, “Primitive Data Definitions”, on page 66.		N/A
StartingSub	A DN with which to start. If none is provided, starts with the first subscriber in the switch.	See definition of ServiceId in Section 5.4, “Primitive Data Definitions”, on page 66.		“”
Max Subscribers	The maximum number of result entries to return.	Int	1 to 40	40
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			N/A

Table 379 Get Keyset Primary List Input Fields

Table 380 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
TotalSubs Found	The number of keyset primary lines that met the search criteria.	Int	0 to 40	N/A
Keyset PrimaryId	The DN of a keyset primary line that met the search criteria (an array of up to 40 of these returned).	See definition of ServiceId in Section 5.4, “Primitive Data Definitions”, on page 66.		N/A
PaginatedInfo	Standard paging output structure. For details, refer to Section 5.5, “Paging”, on page 72.			

Table 380 Get Keyset Primary List Output Fields

9.6 Delete Subscriber

The DeleteSubscriber method is used to remove the subscriber entry. The subscriber may optionally have been disconnected prior to calling this routine by calling the DisconnectSubscriber method.

This method removes any aliases, lines, services and other data that are associated with the DN. If there is a failure, it will continue and delete everything that it can (in case the tables are inconsistent). It will set the HomeDn entry to vacant and remove any existing profile or services data for the subscriber.

If a subscriber is set to be the callback number of a BG subnet, any attempt to delete the subscriber fails. Additionally any attempt to remove a DN number which is used as a default callback number in a Bg subnet is prohibited.

The DeleteSubscriber method may return warnings if it encountered problems.

Table 381 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
1.0.1	N/A	subRC_APP_SUB_LEN_INVALID subRC_APP_SUB_DN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_SUB_DN_IN_USE subRC_APP_SUB_DN_DOES_NOT_EXIST subRC_WAR_SUB_TEEN_DELETION_FAILED subRC_TEEN_LINE_CONFLICT subRC_APP_SUB_MLHG_OWNER_CANNOT_BE_DELETED subRC_APP_SUB_MAIN_NUM_CANNOT_BE_DELETED subRC_APP_SUB_EMERGENCY_NUM_CANNOT_BE_DELETED subRC_APP_BG_ATT_NUM_DELETION_FAILED subRC_WAR_SUB_DELETION_FAILED subRC_SUCCESS

Table 381 Delete Subscriber Return Codes

Table 382 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
ServiceId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required

Table 382 Delete Subscriber Input Fields

9.7 Update Subscriber Status

The UpdateSubscriberStatus method is used to change the status of a subscriber.

Table 383 lists the input fields for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
1.0.1	N/A	subRC_APP_SUB_LEN_INVALID subRC_APP_SUB_DN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_DB_SUB_OFFICE_CODE_LOOKUP_FAILED subRC_DB_SUB_HOME_DN_LOOKUP_FAILED subRC_APP_SUB_STATUS_INVALID subRC_APP_SUB_DN_DOES_NOT_EXIST subRC_DB_SUB_DN_LOOKUP_FAILED subRC_APP_SUB_TEEN_UPDATE_STATUS_NOT_ALLOWED subRC_APP_FEAT_HD_HOT_DESKING_INPROGRESS subRC_DB_SUB_MODIFY_FAILED subRC_SUCCESS

Table 383 Update Subscriber Status Return Codes

Table 384 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
ServiceId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
Status	The new administrative status of the subscriber	enum	Active, Blocked	required

Table 384 Update Subscriber Status Input Fields

9.8 Update Subscriber Account Mgt Info

The UpdateSubscriberAccountMgtInfo method changes the management information of the subscriber's basic profile.

Table 385 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
V5.00.20.01	N/A	subRC_APP_SUB_LEN_INVALID subRC_APP_SUB_DN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_SUB_DN_DOES_NOT_EXIST subRC_DB_SUB_DN_LOOKUP_FAILED subRC_APP_SUB_TEEN_UPDATE_ACCOUNT_MGT_NOT_ALLOWED subRC_APP_FEAT_HD_MODIFICATION_NOT_ALLOWED subRC_APP_SUB_ACCOUNT_LEN_INVALID subRC_APP_SUB_BILLING_ID_LEN_INVALID subRC_APP_SUB_BILLING_ID_INVALID subRC_DB_SUB_XDM_COS_LOOKUP_FAILED subRC_APP_SUB_RETAILER_ID_LEN_INVALID subRC_APP_SUB_XDM_COS_DOES_NOT_EXIST subRC_DB_SUB_XDM_RATE_AREA_LOOKUP_FAILED subRC_APP_SUB_XDM_RATE_AREA_DOES_NOT_EXIST subRC_DB_SUB_XDM_CALL_LOC_LOOKUP_FAILED subRC_APP_SUB_XDM_CALL_LOC_DOES_NOT_EXIST subRC_APP_SUB_EXTERNAL_DN_FLAG_INVALID subRC_DB_SUB_MODIFY_FAILED subRC_APP_FEATURE_PROFILE_OSMO_CONFLICT subRC_APP_NOT_ALLOWED_SETTING_CPU_TO_MOBILEDN subRC_SUCCESS

Table 385 Update Subscriber Account Mgt Info Return Codes

Table 386 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
ServiceId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
AccountId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
BillingId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
RetailerId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
ClassOfService	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A

Table 386 Update Subscriber Account Mgt Info Input Fields (Sheet 1 of 3)

Field Name	Description	Length	Range	Default Value
RateArea	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
TestLineFlag	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
ExternalDNFlag	Indicates that the subscriber DN is a Direct Inward Dialing (DID) number. The subscriber can receive direct incoming calls. If the External DN Flag is set to False, the BGL has a private (non- external) number. This means the number cannot be dialed from outside of the BG and is not displayed as a calling party on calls directed outside the BG. Note that this flag is also used by the MLHG service when an MLHG member originates a call. If True, the terminal’s own data is used as the calling party; otherwise, the Pilot DN’s data is used.	boolean	True,false	N/A
CountryCodeLen	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
CallingLoc	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
FeatureProfileId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
NumberPlanName	The name of the Private Numbering Plan to associate with this BGL. In order to remove an existing Number Plan Name assignment, the parameter can be sent as an empty string. Thereafter, the BG’s numbering plan applies. If the parameter is not sent, no change is made to the existing assignment,	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		N/A
ExternalCallerID		Char array	15 (1…15)	N/A

Table 386 Update Subscriber Account Mgt Info Input Fields (Sheet 2 of 3)

Field Name	Description	Length	Range	Default Value
UseMainPilotDNInternalCalls	Optional. Set 1 means that main pilot DN is used for internal calls. When set 0 means that main pilot DN is not used for internal calls.	boolean	True False	false
UseMainPilotDNExternalCalls	Optional. Set 1 means that main pilot DN is used for external calls. When set 0 means that main pilot DN is not used for external calls.	boolean	True False	false
ImpactLevel	The security zone of a subscriber. V7.00.15.01	enum	ImpactLevelZero, ImpactLevelOne, ImpactLevelTwo, ImpactLevelThree, ImpactLevelFour,> ImpactLevelFive, ImpactLevelSix, ImpactLevelSeven, ImpactLevelEight ImpactLevelNine ImpactLevelUnclassified	

Table 386 Update Subscriber Account Mgt Info Input Fields (Sheet 3 of 3)

9.9 Update Subscriber BG Info

The UpdateSubscriberBGInfo method enables the user to change the BG Info related to the subscriber.

Table 387 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.07.01	11.00.01.01	subRC_APP_SUB_LEN_INVALID subRC_APP_SUB_DN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_LINE_NAME_LEN_INVALID subRC_APP_BG_EXTENSION_LEN_INVALID subRC_APP_BG_DEPT_NAME_LEN_INVALID subRC_APP_SUB_DN_DOES_NOT_EXIST subRC_DB_SUB_DN_LOOKUP_FAILED subRC_APP_SUB_TEEN_UPDATE_ACCOUNT_MGT_NOT_ALLOWED subRC_APP_FEAT_HD_MODIFICATION_NOT_ALLOWED subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DEPT_BG_NAME_MISSING subRC_APP_BG_DEPT_NAME_BG_NAME_CONFLICT subRC_APP_BG_CPU_ID_BG_ID_MISSING subRC_APP_BG_LINE_NAME_BG_NAME_MISSING subRC_APP_BG_EXTENSION_BG_NAME_MISSING subRC_APP_BG_ATT_NUM_BG_NAME_MISSING subRC_APP_BG_DEPT_BG_NAME_MISSING subRC_APP_BG_CPU_ID_OUT_OF_RANGE subRC_DB_BG_CPU_ID_NUMBER_LOOKUP_FAILED subRC_APP_BG_CPU_ID_NUMBER_OUT_OF_RANGE subRC_DB_BG_NUMBER_CPU_LOOKUP_FAILED subRC_APP_BG_CPU_SUB_OUT_OF_RANGE subRC_APP_BG_ATT_NUM_DELETION_FAILED subRC_DB_SUB_INSERTION_FAILED subRC_SUCCESS subRC_APP_UNICODE_BG_LINE_NAME_LEN_INVALID

Table 387 Update Subscriber BG Info Return Codes

Table 388 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
ServiceId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
BGName	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
BGLineName	The subscriber member name.	20	Character string	“”
UnicodeBGLineName	A BG Line name for use in markets such as China where multi-byte characters are allowed.	1..24	Wide character string.	“”

Table 388 Update Subscriber BG Info Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
BGAttendantNumber	Indicates the DN is used for the Attendant Number function of the BG.	Boolean	True, False	False
CpuId	A Call Pickup group Id to which this subscriber (BGL) belongs. There can be only one CPU group ID per subscriber. The CfCPU service must be provisioned either by explicit assignment or Feature Profile inheritance in order for the CPU service to be effective. A value of 0 indicates that the subscriber is not part of any CPU.	Int	1-10000	0 – no CPU Id
CpuName	Call Pickup group name		Character string	
CpuNotificationDelayTimer	Notification Timer for Call Pick Up	Int	0…60	0
Extension	This field indicates the Extension number of the subscriber (BGL It is an optional field and being used for intra BG calls display only.	7	Numeric string	“”
BGDeptName	The department name associated with the Subscriber.	20	Character string	“”

Table 388 Update Subscriber BG Info Input Fields (Sheet 2 of 2)

9.10 Update Subscriber Account User Info

The UpdateSubscriberAccountUserInfo method changes the user-controlled information of the subscriber's basic profile.

Table 389 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
1.0.1	N/A	subRC_APP_SUB_LEN_INVALID subRC_APP_SUB_DN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_SUB_DN_DOES_NOT_EXIST subRC_DB_SUB_DN_LOOKUP_FAILED subRC_APP_SUB_ACC_USER_INFO_TEEN_NOT_ALLOWED subRC_APP_FEAT_HD_HOT_DESKING_INPROGRESS subRC_APP_SUB_USER_INFO_DISPLAY_NAME_OUT_OF_RANGE subRC_APP_SUB_USER_INFO_TIME_ZONE_LEN_INVALID subRC_APP_SUB_USER_INFO_TIME_ZONE_INVALID subRC_DB_SUB_USER_INFO_TIME_ZONE_LOOKUP_FAILED subRC_APP_SUB_ACC_USER_INFO_PIN_LEN_INVALID subRC_APP_SUB_ACC_USER_INFO_PIN_INVALID subRC_APP_SUB_HD_PIN_CONFLICT subRC_APP_UTIL_LANG_LEN_INVALID subRC_APP_UTIL_LANG_NOT_ALLOWED subRC_APP_UTIL_LANG_INVALID subRC_DB_SUB_MODIFY_FAILED subRC_SUCCESS subRC_APP_SUB_USER_INFO_UNICODE_DISPLAY_NAME_OUT_OF_RANGE

Table 389 Update Subscriber Account User Info Return Codes

Table 390 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
ServiceId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. Note that for this request, if the OperatorId is equal to the ServiceId, this is treated as an indication that the request is from the iSSC (Subscriber Self-Care), and thus the subscriber’s profile is not updated to reflect the time of the update and the ID of the operator, since these fields are intended to record updates by operators other than the end user (subscriber).			required
DisplayName	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A

Table 390 Update Subscriber Account User Info Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
UnicodeDisplayName	A display name for use in markets such as China where multi-byte characters are allowed.	1..24	Character string.	“”
TimeZone	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
Pin1	Please refer to the definition of Pin in Section 9.1, “Create Subscriber”, on page 303.			N/A
Pin2	Please refer to the definition of Pin in Section 9.1, “Create Subscriber”, on page 289.			N/A
Pin3	Please refer to the definition of Pin in Section 9.1, “Create Subscriber”, on page 289.			N/A
Pin4	Please refer to the definition of Pin in Section 9.1, “Create Subscriber”, on page 289.			N/A
Pin5	Please refer to the definition of Pin in Section 9.1, “Create Subscriber”, on page 289.			N/A
PublicPin	Please refer to the definition of PublicPin in Section 9.1, “Create Subscriber”, on page 303. Note: added in WSDL version 10.00.08.01.			N/A
Language	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
BusinessAuthCode	Subscriber can make Business call	char	2…14	
PrivateAuthCode	Subscriber can make private call	char	2…14	

Table 390 Update Subscriber Account User Info Input Fields (Sheet 2 of 2)

9.11 Update Subscriber Capabilities

This method changes the subscriber line capabilities.

Table 391 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
1.0.1	N/A	subRC_APP_COMM_OPERATOR_INVALID subRC_APP_SUB_LEN_INVALID subRC_APP_SUB_DN_INVALID subRC_APP_SUB_DN_DOES_NOT_EXIST subRC_DB_SUB_DN_LOOKUP_FAILED subRC_APP_SUB_UPDATE_CAP_TEEN_NOT_ALLOWED subRC_APP_FEAT_HD_HOT_DESKING_INPROGRESS subRC_APP_SUB_UPDATE_CAP_VOIP_MISSING subRC_APP_SUB_T38_ECH_CAN_CONFLICT subRC_DB_SUB_MODIFY_FAILED subRC_SUCCESS

Table 391 Update Subscriber Capabilities Return Codes

Table 392 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
ServiceId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
Bearer_Speech	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
Bearer_3_1K	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
Bearer_56K	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
Bearer_64K	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
FaxMode	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
EchoCancellation	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
SilenceSuppression	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
FaxStatistics	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A

Table 392 Update Subscriber Capabilities Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
OutbandDTMFSignaling	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
SuppressOutbandDTMFWithG711	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A

Table 392 Update Subscriber Capabilities Input Fields (Sheet 2 of 2)

9.12 Update Keyset Info

The UpdateKeysetInfo method is used to update the keyset-related data for a keyset subscriber. Data that can be modified includes: KeysetUse (None, Primary, or Phantom); KeysetOperationMode (DeviceBased or LineBased); ManualHoldTimer; and KeysetLineAppearances (full list must be provided).

Table 393 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.01.01 05.04.02[a] 10.00.01.34[b] V5.00.20.01	N/A	subRC_SUCCESS subRC_APP_SUB_LEN_INVALID subRC_APP_SUB_DN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_SUB_DN_DOES_NOT_EXIST subRC_DB_SUB_DN_LOOKUP_FAILED subRC_DB_SUB_OFFICE_CODE_LOOKUP_FAILED subRC_DB_SUB_HOME_DN_LOOKUP_FAILED subRC_DB_XDM_EP_LOOKUP_FAILED subRC_APP_EP_TYPE_INVALID subRC_APP_SUB_KEYSET_USE_MISSING subRC_APP_SUB_KEYSET_USE_INVALID subRC_APP_SUB_KEYSET_REG_TYPE_INVALID subRC_APP_SUB_KEYSET_DATA_MISSING subRC_DB_SUB_KEYSET_LOOKUP_FAILED subRC_DB_SUB_KEYSET_LINE_DATA_INCONSISTENCY subRC_APP_SUB_KEYSET_LINE_APP_CONFLICT subRC_APP_SUB_KEYSET_CALL_FRW_LED_IND_CONFLICT subRC_APP_SUB_KEYSET_MANUAL_HOLD_TIMER_INVALID subRC_APP_SUB_KEYSET_LINE_NUMBER_OUT_OF_RANGE subRC_APP_SUB_KEYSET_LINE_INVALID subRC_APP_SUB_KEYSET_LINE_DOES_NOT_EXIST subRC_DB_SUB_KEYSET_LINE_LOOKUP_FAILED subRC_APP_SUB_KEYSET_LINE_APP_NOT_KEYSET subRC_APP_SUB_KEYSET_LINE_APP_BG_CONFLICT subRC_APP_SUB_KEYSET_LINE_ZERO_LENGTH_INVALID subRC_APP_SUB_KEYSET_SUB_NUM_OUT_OF_RANGE subRC_DB_SUB_KEYSET_INSERTION_FAILED subRC_DB_COMM_INTERNAL_RECOVERY_ERROR subRC_DB_SUB_KEYSET_MODIFY_FAILED subRC_DB_COMM_INTERNAL_RECOVERY_ERROR subRC_APP_SUB_KEYSET_MANUAL_HOLD_TIMER_OUT_OF_RANGE subRC_DB_SUB_KEYSET_INFO_DELETION_FAILED subRC_APP_SUB_KEYSET_SHARED_MEM_FAILED subRC_DB_EP_XDM_DELETION_FAILED subRC_DB_EP_XDM_CREATE_FAILED subRC_DB_SUB_XDM_ASSIGN_FAILED subRC_APP_KEYSET_OSMO_CONFLICT
[a]Added KeysetUse as a modifiable parameter. [b]Added CallForwardLEDIndication.		

Table 393 Update Keyset Info Return Codes

Table 394 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
KeysetUse	Describes whether a DN is to be used for Keyset Operation and, if so, in what way. For descriptions of the possible values, refer to Section 4.13, “Keyset Information”, on page 55.	enum	KeysetPrimaryLine KeysetPhantomLine KeysetNone	N/A
KeysetManualHoldTimer	Activated when the user places the line on manual hold. If this line is picked up on any device where this DN has an appearance, the timer is stopped. If the timer expires, the device on which the call was placed on hold is recalled.	integer	5..3600 (seconds)	40[1]
KeysetLine Appearances	Applicable only when KeysetUse is KeysetPrimaryLine. A list of DNs to be associated as secondary lines on the device where the subscriber/BGL being updated is the KeysetPrimaryLine. Along with each DN is an Originating Operation Mode and a Terminating Operation Mode. Refer to Section 4.13, “Keyset Information”, on page 55 for more information.	1..630	See Section 4.13, “Keyset Information”, on page 55.	N/A
CallForward LEDIndication	Indicates whether the line key LEDs on the device where this subscriber is the Keyset Primary shall wink whenever those lines have call forwarding active. This parameter is optional when creating a subscriber and when updating keyset info for a subscriber.	Boolean	True, False	True[1]
1The default value for this parameter applies only to the CreateSubscriber SOAP request. For the UpdateKeysetInfo SOAP request, the default is not used. If the tag is sent, the value is applied; if not, the existing value is left unchanged.				

Table 394 Update Keyset Info Input Fields

9.13 Update Subscriber Dn

The UpdateSubscriberDn method is used to change the DN of a subscriber. The new DN must have been defined as available on the switch and Available (vacant).

Calls to the old number will produce either a disconnect or NewNumber announcement. A flag is used to control the announcement, because the new number may be nonpublished.

If a subscriber is changed to a DN number that already exists as a callback number of a BG subnet, any attempt to delete the subscriber fails. Additionally any attempt to change a DN number which is used as a default callback number in a Bg subnet is prohibited.

See subscriber interactions for potential consequences of using this method.

Table 395 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
06	N/A	subRC_SUCCESS
		subRC_DB_COMM_INTERNAL_RECOVERY_ERROR
		subRC_APP_SUB_LEN_INVALID
		subRC_APP_SUB_DN_INVALID
		subRC_APP_COMM_OPERATOR_INVALID
		subRC_APP_SUB_DN_EXIST
		subRC_DB_SUB_DN_LOOKUP_FAILED
		subRC_APP_SUB_TEEN_UPDATE_CONNECTION_NOT_ALLOWED
		subRC_APP_SUB_DN_DOES_NOT_EXIST
		subRC_DB_SUB_KEYSET_LOOKUP_FAILED
		subRC_APP_FEAT_HD_MODIFICATION_NOT_ALLOWED
		subRC_APP_SUB_MLHG_OWNER_CONFLICT
		subRC_APP_SUB_MLHG_MEMBER_CONFLICT
		subRC_APP_SUB_OFFICE_CODE_DOES_NOT_EXIST
		subRC_DB_SUB_OFFICE_CODE_LOOKUP_FAILED
		subRC_APP_SUB_HOME_DN_DOES_NOT_EXIST
		subRC_DB_SUB_HOME_DN_LOOKUP_FAILED
		subRC_DB_XDM_EP_LOOKUP_FAILED
		subRC_DB_SUB_FEATURE_LOOKUP_FAILED
		subRC_DB_SUB_FEATURE_MODIFY_ERROR
		subRC_DB_SUB_MODIFY_FAILED
		subRC_DB_EP_SIP_LOOKUP_FAILED

Table 395 Update Subscriber Dn Return Codes (Sheet 1 of 2)

WSDL Version	Obsolete in WSDL Version	Return Codes
		subRC_DB_EP_SIP_INSERTION_FAILED
		subRC_DB_SUB_KEYSET_INSERTION_FAILED
		subRC_DB_COMM_RTM_LOOKUP_FAILED
		subRC_DB_COMM_RTM_DELETION_FAILED
		subRC_DB_COMM_RTM_CREATION_FAILED
		subRC_DB_COMM_XDM_ALIAS_LOOKUP_FAILED
		subRC_DB_EP_XDM_DELETION_FAILED
		subRC_DB_EP_XDM_CREATE_FAILED
		subRC_DB_XDM_EP_LOOKUP_FAILED
		subRC_DB_EP_XDM_CREATE_FAILED
		subRC_DB_SUB_XDM_ASSIGN_FAILED
		subRC_INTERNAL_ERROR
		subRC_APP_SUB_NOT_BELONG_BG
		subRC_APP_FEAT_CPS_REQUIRES_CT
		subRC_APP_FEAT_CPS_INVALID_CPS_SETTINGS
		subRC_APP_LOTID_ASSIGN_TO_FP_FAILURE
		subRC_APP_LOTID_UNASSIGN_FROM_FP_FAILURE
		subRC_APP_LOTID_CHANGE_IN_FP_FAILURE
		subRC_APP_LOTID_RETRIEVE_FAILURE

Table 395 Update Subscriber Dn Return Codes (Sheet 2 of 2)

Table 396 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
ServiceId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
InterceptAnnouncement	The intercept announcement to which calling parties are routed after this subscriber’s number is changed.	Enum	None, Disconnect, NumberChange	N/A
NewServiceId	This field is used when a subscriber’s ServiceId has changed. It may optionally be played during the NumberChange announcement.	See definition of ServiceId in Section 5.4, “Primitive Data Definitions”, on page 66.		required

Table 396 Update Subscriber Dn Input Fields

9.14 Update Subscriber Features

The UpdateSubscriberFeatures method is used to add, edit, remove, or deny features to a subscriber.

Table 397 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes	Comments
1.0.1	N/A	subRC_SUCCESS	
		subRC_APP_SUB_LEN_INVALID	
		subRC_APP_SUB_DN_INVALID	
		subRC_APP_COMM_OPERATOR_INVALID	
		subRC_APP_SUB_DN_DOES_NOT_EXIST	
		subRC_DB_SUB_DN_LOOKUP_FAILED	
		subRC_APP_FEAT_HD_HOT_DESKING_INPROGRESS	
		subRC_DB_SUB_FEAT_XDM_RCF_ERROR	
		subRC_APP_FEAT_RCF_CONFLICT	
		subRC_APP_FEAT_BGCNAM_DENY_NOT_ALLOWED	
		subRC_APP_FEAT_BGCNAM_MODIFY_NOT_ALLOWED	
		subRC_APP_FEAT_BGCNAM_NAMEDB_SCP_INVALID	
		subRC_APP_SUB_NOT_BELONG_BG	
		subRC_APP_FEAT_BGCT_DENY_NOT_ALLOWED	
		subRC_APP_FEAT_BGCT_MODIFY_NOT_ALLOWED	
		subRC_APP_FEAT_BGDAPPS_DENY_NOT_ALLOWED	
		subRC_APP_FEAT_BGDAPPS_MODIFY_NOT_ALLOWED	
		subRC_APP_FEAT_BGDAPPS_INTA_BG_DISPLAY_CONFLICT	
		subRC_APP_FEAT_BGDNPPS_DENY_NOT_ALLOWED	
		subRC_APP_FEAT_BGDNPPS_MODIFY_NOT_ALLOWED	
		subRC_APP_FEAT_BGDNPPS_PPS_CALLER_ID_CONFLICT	
		subRC_APP_FEAT_BGDNPPS_CALLER_ID_INVALID	
		subRC_SYS_UTIL_GET_TIME_FAILED	
		subRC_APP_FEAT_BGCISNAME_NAMEDB_SCP_INVALID	
		subRC_APP_FEAT_CNAM_DENY_NOT_ALLOWED	
		subRC_APP_FEAT_CNAM_MODIFY_NOT_ALLOWED	
		subRC_APP_FEAT_CT_DENY_NOT_ALLOWED	
		subRC_APP_FEAT_DAPPS_DENY_NOT_ALLOWED	
		subRC_APP_FEAT_DAPPS_MODIFY_NOT_ALLOWED	
		subRC_APP_FEAT_DNPPS_DENY_NOT_ALLOWED	
		subRC_APP_FEAT_ECF_DAY_INVALID	

Table 397 Update Subscriber Features Return Codes (Sheet 1 of 3)

WSDL Version	Obsolete in WSDL Version	Return Codes	Comments
		subRC_APP_FEAT_ECF_START_TIME_INVALID subRC_APP_FEAT_ECF_STOP_TIME_INVALID subRC_APP_FEAT_ECF_START_STOP_TIME_CONFLICT subRC_APP_FEAT_ECF_TIME_RANGE_OVERLAPPED subRC_APP_FEAT_IN_PROFILE_NAME_DOES_NOT_EXIST subRC_DB_FEAT_IN_REF_COUNT_MODIFY_FAILED subRC_DB_FEAT_PRES_MODIFY_FAILED subRC_APP_FEAT_SPCALL1_DN_INVALID subRC_APP_FEAT_SPCALL1_CODE_INVALID subRC_APP_FEAT_SPCALL2_DN_INVALID subRC_APP_FEAT_SPCALL2_CODE_INVALID subRC_APP_FEAT_TRS_RESTRICTED_DN_INVALID subRC_VMWI_NOTIFY_FAILURE subRC_APP_FEAT_VM_NOTIFY_FAILED subRC_APP_FEAT_HD_HOT_DESKING_INPROGRESS subRC_APP_FEAT_HD_LOOKUP_FAILED subRC_APP_FEAT_HD_SIDE_NOT_ALLOWED subRC_APP_FEAT_HD_ROUTING_ERROR subRC_APP_FEAT_HD_REMOTE_ASSIGN_NOT_ALLOWED subRC_APP_FEAT_SCC_ASSIGN_NOT_ALLOWED subRC_APP_FEAT_SCC_BG_LARGE_CONFERENCE_REQUIRED	
V4.00.04.01		subRC_APP_SUB_NOT_BELONG_BG	30100006 Attempt to assign CPS feature, but subscriber not part of BG
		subRC_APP_FEAT_CPS_REQUIRES_CT	31158000 Feature CPS require CT.
		subRC_APP_FEAT_CPS_INVALID_CPS_SETTINGS	31158001 Setting both Parking Allowed and Retrieval Allowed to false is forbidden
		subRC_APP_LOTID_ASSIGN_TO_FP_FAILURE	33000019 Failure to assign lot index to feature profile

Table 397 Update Subscriber Features Return Codes (Sheet 2 of 3)

WSDL Version	Obsolete in WSDL Version	Return Codes	Comments
		subRC_APP_LOTID_UNASSIGN_FROM_FP_FAILURE	33000020 Failure to unassign lot index from feature profile
		subRC_APP_LOTID_CHANGE_IN_FP_FAILURE	33000021 Failure to change lot index to feature profile
		subRC_APP_LOTID_RETRIEVE_FAILURE	33000022 Failure to get Lot Data Entry

Table 397 Update Subscriber Features Return Codes (Sheet 3 of 3)

Table 398 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
ServiceId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. Note that for this request, if the OperatorId is equal to the ServiceId, this is treated as an indication that the request is from the iSSC (Subscriber Self-Care), and thus the subscriber’s profile is not updated to reflect the time of the update and the ID of the operator, since these fields are intended to record updates by operators other than the end user (subscriber).			required
ListOfFeatures	The list of features to be modified.		A list of pointers to the data structures of all the features to be modified. The supported features are discussed in Chapter 10, “Features”	N/A

Table 398 Update Subscriber Features Input Fields

9.15 Get Sub Tran Status

The Get Sub Tran Status SOAP request is used to retrieve the transient status of a subscriber. The information about the current status of the specified subscriber is obtained from the PDM (Port Data Manager) component.

Table 399 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
11.00.12.03	N/A	subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_SUB_LEN_INVALID subRC_APP_SUB_DN_INVALID subRC_SYS_RTP_CONNECTION_ERROR subRC_APP_SUB_PDM_TRAN_STATUS_RESPONSE_ERROR subRC_APP_SUB_PDM_TRAN_STATUS_TOKEN_ERROR

Table 399 Get Sub Tran Status Return Codes

Table 400 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Table 10 in Section 5.4.2, “Field Definitions” .		required
ServiceId	The DN for which the status is being requested.	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		required

Table 400 Get Sub Tran Status Input Fields

Table 401 lists the output fields for this method.

Field Name	Description	Length	Range	Default Value
ServiceId	The DN for which the status has been requested.	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		N/A
CallLegDataList	A list of CallLegDataEntry elements, each consisting of the following attributes:			
(CallLegData Entry) Id	An integer value identifying the call leg to which this data entry element applies.	integer	1..max number of possible call legs	N/A
(CallLegData Entry) Status	An enumerated value describing the current state of the call.	enum	Idle, BLV_Active, CallSetupOutgoing, Ringing, BusyOutgoing, BusyIncoming, Releasing	N/A

Table 401 Get Sub Tran Status Output Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
(CallLegData Entry) OriginalCalled PartyDn	The DN of the original called party for this call leg.	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		N/A
(CallLegData Entry) CaliingPartyDn	The DN of the calling party for this call leg.	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		N/A

Table 401 Get Sub Tran Status Output Fields (Sheet 2 of 2)

9.16 Update Connection Info

The UpdateConnectionInfo method is used to change the subscriber's connection information. Not all fields can be updated. The Connection information is SIP. Table 402 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
1.0.1	N/A	subRC_SUCCESS subRC_DB_COMM_INTERNAL_RECOVERY_ERROR subRC_APP_SUB_LEN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_SUB_DN_DOES_NOT_EXIST subRC_DB_SUB_DN_LOOKUP_FAILED subRC_DB_SUB_OFFICE_CODE_LOOKUP_FAILED subRC_APP_EP_TYPE_INVALID subRC_APP_EP_DISCONNECTED subRC_DB_COMM_RTM_LOOKUP_FAILED subRC_DB_COMM_RTM_MODIFY_FAILED subRC_APP_SUB_MLHG_OWNER_CONFLICT subRC_APP_SUB_MLHG_MEMBER_CONFLICT subRC_DB_SUB_XDM_UNASSIGN_FAILED subRC_APP_EP_IP_LEN_INVALID subRC_APP_EP_IP_INVALID subRC_DB_EP_MODIFY_FAILED subRC_APP_EP_SIP_CONNECTION_INFO_MISSING subRC_APP_EP_SIP_DYNAMIC_IP_CONFLICT subRC_APP_EP_SIP_STATIC_IP_PORT_MISSING subRC_APP_EP_IP_LEN_INVALID subRC_APP_EP_PRE_REG_TYPE_DYNAMIC_CONFLICT subRC_APP_EP_IPPHONE_MAX_SESSIONS_INVALID subRC_APP_EP_SIP_SCHEME_SECURITY_CONFLICT subRC_APP_EP_SIP_REALM_LEN_INVALID subRC_APP_EP_SIP_USER_NAME_LEN_INVALID subRC_APP_EP_SIP_PASSWORD_LEN_INVALID subRC_APP_EP_SIP_SECURITY_NUMBER_OUT_OF_RANGE subRC_DB_EP_SIP_SECURITY_MODIFY_FAILED subRC_APP_EP_PROFILE_LEN_INVALID

Table 402 Update Connection Info Return Codes

Table 403 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
ServiceId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
IPPhoneUpdateInfo	Please refer to ConnectionInfo (IPPhone) definitions in Section 9.1, “Create Subscriber”, on page 303			

Table 403 Update Connection Info Input Fields

9.17 Update Subscriber Feature Profile

This method associates a subscriber with a feature profile. To remove a subscriber’s association with a feature profile, specify a blank FeatureProfileName and do not specify a FeatureProfile BgName.

Table 404 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01.12		subRC_SUCCESS subRC_APP_SUB_LEN_INVALID subRC_APP_SUB_DN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_SUB_DN_DOES_NOT_EXIST subRC_DB_SUB_DN_LOOKUP_FAILED subRC_APP_SUB_TEEN_UPDATE_ACCOUNT_MGT_NOT_ALLOWED subRC_APP_FEAT_HD_MODIFICATION_NOT_ALLOWED subRC_APP_FP_NAME_LEN_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_FP_NAME_MISSING subRC_DB_SUB_FP_RESTORE_ORIGINAL_FAILED subRC_APP_SUB_FP_UPDATE_FAILED subRC_DB_SUB_MODIFY_FAILED

Table 404 Update Subscriber Feature Profile Return Codes

Table 405 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
ServiceId	The subscriber whose association to a feature profile is being altered.	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		required
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required

Table 405 Update Subscriber Feature Profile Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
PreserveLoc Pres	If set and the new feature profile has some of the same service features as the old feature profile, the subscriber's local settings will be maintained, otherwise, if false, the subscriber's settings for the features inherited from the feature profile, will be set to the values from the feature profile.	Boolean	True, False	False
FeatureProfile ID	Identifies the Feature Profile; comprises the Feature Profile Name and the BGName.			
(FeatureProfile ID) Feature Profile Name	The name of the feature profile	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.		required
(FeatureProfile ID) BG Name	The name of the business group with which the feature profile is associated.	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.		""
Comments:	Update the subscriber to the New Feature Profile. An empty tag of the Feature Profile Name and BGName means to un-assign the subscriber from the Feature Profile.			

Table 405 Update Subscriber Feature Profile Input Fields (Sheet 2 of 2)

9.18 Add Contact List

This method adds a contact to a SIP connection.

Table 406 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01.04		subRC_SUCCESS subRC_APP_SUB_DN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_SUB_DN_DOES_NOT_EXIST subRC_DB_SUB_DN_LOOKUP_FAILED subRC_DB_XDM_EP_LOOKUP_FAILED subRC_APP_EP_CONTACT_INVALID subRC_APP_EP_CONTACT_OUT_OF_RANGE subRC_APP_EP_CONTACT_ADDR_LEN_INVALID subRC_APP_EP_CONTACT_ADDR_INVALID subRC_APP_EP_CONTACT_PORT_MISSING subRC_APP_EP_CONTACT_PORT_INVALID subRC_APP_EP_CONTACT_TYPE_MISSING subRC_APP_EP_CONTACT_TYPE_INVALID subRC_APP_EP_CONTACT_MODIFY_FAILED

Table 406 AddContactList Return Codes

Table 407 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
ServiceId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
HostAddr	Host address for Static SIP subscriber.	255	Valid host address	
Port	Signaling port of the host address.	Int	Valid Port value	
IPConnection Type	IPConnectionType – transport type	Enum	UDP, TCP, TLS	UDP
MediaType	Mediatype. Display only.	Enum	mtAudio, mtVideo, mtInstantMessage	mtAudio
Priority	Priority. Not supported yet.	Int		
TimeToLive	Display only. Refer to Section 9.1, “Create Subscriber”, on page 303.	Int	(-1..2147483647) -1 = unregistered 0 = static registered (never expired)	-1
ExpirationTime	Display only.	xsd_dateTime	TimeToLive in date and time format.	N/A
Comments:	The contact list is allowed only to provision for SIP static subscriber.			

Table 407 AddContactList Input Fields

9.19 Delete Contact List

This method deletes a contact list.

Table 408 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01.24		subRC_SUCCESS subRC_APP_SUB_DN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_SUB_DN_DOES_NOT_EXIST subRC_DB_SUB_DN_LOOKUP_FAILED subRC_DB_XDM_EP_LOOKUP_FAILED subRC_APP_EP_CONTACT_INVALID subRC_APP_EP_CONTACT_MODIFY_FAILED

Table 408 Delete Contact List Return Codes

Table 409 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
ServiceId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
HostAddr	Host address for static SIP subscriber.	255	Valid host address	
Port	Signaling port of the host address.	Int	Valid Port value	
IPConnection Type	IPConnectionType – transport type.	Enum	UDP, TCP, TLS	UDP
MediaType	Mediatype. Display only.	Enum	mtAudio, mtVideo, mtInstantMessage	mtAudio
Priority	Priority. Not supported yet.	Int		
TimeToLive	Display only. Refer to Section 9.1, “Create Subscriber”, on page 303.	Int	(-1..2147483647) -1 = unregistered 0 = static registered (never expired)	-1
ExpirationTime	Display only.	xsd_dateTime	TimeToLive in date and time format.	N/A
Comments:	The contact list is allowed only to provision for SIP static subscriber.			

Table 409 Delete Contact List Input Fields

9.20 Get Contact List

This method retrieves a contact list.

Table 410 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01.24		subRC_SUCCESS subRC_APP_SUB_DN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_SYS_EP_CONTACT_ALLOC_ERR subRC_DB_EP_XDM_CONTACT_LOOKUP_FAILED

Table 410 Get Contact List Return Codes

Table 411 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
ServiceId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
Paging	Paging structure for Get List requests. For more information, see Section 5.5, “Paging”, on page 72.			
InterfaceWSDL	Refer to Section 5.1.1, “The Interface WSDL and Interface Release Control”, on page 61.			

Table 411 Get Contact List Input Fields

9.21 Update Mobile Info

This method updates mobile info.

Table 412 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
V5.00.20.01		subRC_APP_NOT_ALLOWED_SETTING_KEYSET_TO_MOBILEDN subRC_APP_FEAT_OSMO_ONS_NOT_ASSIGNED_TO_DESK_PHONE_DN subRC_APP_FEAT_OSMO_DESK_PHONE_NUMBER_IN_USE subRC_APP_SUB_MOBL_RING_DUR_INVALID subRC_APP_SUB_MAIN_RING_DUR_INVALID subRC_APP_SUB_CELL_RING_DUR_INVALID subRC_APP_FEATURES_AND_OSMO_CONFLICT subRC_APP_FEATURE_PROFILE_OSMO_CONFLICT subRC_APP_SUB_OSMO_FEATURES_CONFLICT subRC_APP_SUB_OSMO_FP_CONFLICT subRC_APP_SUB_OSMO_CPG_MEMBER_CONFLICT subRC_APP_SUB_OSMO_COM_MEMBER_CONFLICT subRC_APP_SUB_OSMO_EA_MEMBER_CONFLICT subRC_APP_SUB_OSMO_MLHG_MEMBER_CONFLICT subRC_APP_SUB_OSMO_KEYSET_CONFLICT subRC_APP_SUB_OSMO_REMOVAL_FAILURE subRC_APP_OSMO_TLS_CONFLICT subRC_APP_FEAT_OSMO_CSTA_NOT_ASSIGNED_TO_DESK_PHONE_DN

Table 412 Update Mobile Info Return Codes

Table 413 lists the input fields for this method.

Field Name	Description	Required	Length	Range	Default Value
Operator ID	Please refer to Section 5.4, "Primitive Data Definitions", on page 80.				
Service ID	The subscriber to whom will assign/remove OSMO capability	yes	Please refer to Section 5.4, "Primitive Data Definitions", on page 80.		
MobileFlag	Enables OSMO capability	No	Boolean	False true	false
MainDeviceDN	The associated Deskphone	No (yes when MonileFlag = true)	Please refer to Section 5.4, "Primitive Data Definitions", on page 80.		
CellDeviceNumber	Cell phone number of an OSMO	no	string	[1-29] '+' before number is required	NA
RouteToCell	Option to route the call to the mobile phone (cell device number) V7.01.01.01	No	boolean	[false,true] Must not be true when CellDeviceNumber is not defined	NA
MobileDeviceRing Duration	Ring duration for the Mobile subscriber	No	integer	5,10,15 up to V7.00.23.01) 1-25 from V7.00.24.01	10 up to V7.00.23.01) 15 from V7.00.24.01
MainDeviceRingD uration	Ring duration for the Deskphone	No	integer	5,10,15 up to V7.00.23.01) 1-25 from V7.00.24.01	10 up to V7.00.23.01) 15 from V7.00.24.01
CellDeviceRingDu ration	Ring duration for the cellular phone	no	integer	5,10,15 up to V7.00.23.01) 1-25 from V7.00.24.01	10 up to V7.00.23.01) 15 from V7.00.24.01
· OSMO – DeskPhone Synchronization: When assigning OSMO service to a subscriber, the following Subscriber OSMO fields are inherited from tthe DeskPhone subscriber:					
LocalRealm	This identifies the local realm. A realm is a string to be displayed to users for association with its username and password. For a trusted entity, this parameter should not be provisioned. For a nontrusted entity, if this parameter is not provisioned, the default localhost entry information is used.	0..128	Character string	"" OR if subscriber is OSMO, then it is assigned the LocalRealm value of the DeskPhone	LocalRealm

Table 413 Update Mobile Info Input Fields (Sheet 1 of 3)

Field Name	Description	Required	Length	Range	Default Value
LocalUserName	This identifies a user within the local realm. For a trusted entity, this parameter should not be provisioned. For a nontrusted entity, if this parameter is not provisioned, the default localhost entry information is used.	0, 6..64	Character string	“”OR if subscriber is OSMO, then it is assigned the OSMO Subscriber DN value	LocalUserName
LocalPassword	The password for the local user within the local realm. For a trusted entity, this parameter should not be provisioned. For a non-trusted entity, if this parameter is not provisioned, the default localhost entry information is used.	6..20	Character string	“” OR if subscriber is OSMO, then it is assigned the LocalPassword value of the DeskPhone	LocalPassword
BGLineName	Business Group Line Name is the subscriber member name.	0-20	Character string	“” OR if subscriber is OSMO, then it is assigned the LocalRealm value of the DeskPhone	BGLineName
UnicodeBGLineName	A BG Line name for use in markets such as China where multi-byte characters are allowed.	1…24	Wide character string.	“”OR if subscriber is OSMO, then it is assigned the OSMO Subscriber DN value	UnicodeBGLineName

Table 413 Update Mobile Info Input Fields (Sheet 2 of 3)

Field Name	Description	Required	Length	Range	Default Value
DisplayName	Used for in-switch and VPN dialing to present the calling party's name. Used in installations where the SCP is unavailable or the switch is configured to use this field instead of the SCP. This field may be up to 20 characters in length; note, however, that in some networks, ISUP supports only 15 characters. In these environments, only the first 15 characters of the field will be sent. SIP phones may populate a generic name field that contains an alphabetical name to be associated with the number. If populated, that field will be used instead of this field. *Optional*	1…20	Character string	"" OR if subscriber is OSMO, then it is assigned the LocalPassword value of the DeskPhone	DisplayName
UnicodeDisplayName	A display name for use in markets such as China where multi-byte characters are allowed.	1…24	Wide character string.	"" OR if subscriber is OSMO, then it is assigned the LocalPassword value of the DeskPhone	UnicodeDisplayName

Table 413 Update Mobile Info Input Fields (Sheet 3 of 3)

10 Features

This chapter describes the list of the features available on the system for provisioning with the SOAP/XML interface. Only subscribers with the compatible attributes can subscribe to features. For example, **Call Waiting** is not applicable to SIP subscribers and **Busy Line Verification** is not applicable for Enterprise customers. For more detailed descriptive information on OpenScape Voice features, refer to the *OpenScape Voice Feature Description Guide*.

In addition to the “Subscribed” parameter (in the Field Name column in the following sections), features that have attributes will also have a “Local” parameter. The Local parameter consists of a list of local presence flags that apply when a feature is inherited from a feature profile. They are used when displaying data to indicate the attributes that are overwritten locally (at the subscriber/BGL level), and are used when updating data to reset the locally overwritten feature data to the values at the feature profile level. To avoid redundancy, this Local parameter is not listed for every feature. For further information, refer to the definition of “Local” in Table 11 on page 67.

Attention: The acronyms that follow the feature name are the tags used when developing the SOAP client. Some do not match the feature title as the feature names have been modified for the OpenScape Voice environment. Note: Cf = calling feature

---

The features without an “Enabled” attribute default to ‘Active’ when assigned as there is no way to activate them later. Features having an “Enabled” attribute default to inactive when assigned without passing the Enabled attribute, except in a few cases in which the Enabled attribute was not existing and was added later. In these cases, the default remains active in order to avoid changing pre-existing behavior.

10.1 Account Code (CfAcctCode)

The Business Group Account Codes feature lets the subscriber add a number (the account code) into the CDR record for allocation of charges on billable calls (incoming or outgoing).

The system administrator specifies the number of digits of the account code. It can be from 2 to 14 digits long; its length is the same for all stations in a business group.

Table 414 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
Enabled	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False

Table 414 CfAcctCode – Account Code Input and Output Parameters

10.2 Anonymous Caller Rejection (CfACR)

The Anonymous Call Rejection feature provides subscribers the capability to reject calls from parties who have a privacy feature active (such as caller ID blocking) that prevents the delivery of the calling number to the called party.

Table 415 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
Enabled	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False
SwitchControl	Specifies the allowable range of callback destinations. The UseSwitchSetting value indicates that this option is controlled by a switch wide RTP configuration parameter and its setting should be used. The switch wide parameter is Srx/Service/ACAR/ PerSPCSBlkARMsgWICNInfo	enum	IntraSwitch InterSwitch UseSwitchSetting	UseSwitch Setting
Billing	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			FlatRate

Table 415 CfACR – Anonymous Caller Rejection Input and Output Parameters

10.3 Attendant Answering Position (CfBgNBCPU)

This feature may be provisioned for a Business Group and is applicable to the entire business group. It may not be provisioned against individual subscribers or Business Group Lines and the feature-related data is not propagated to the subscriber/BGL level.

One of the possible destinations for the Hunt Group Night Service DN present in the Hunt Group SOAP requests described in Section 15.2, “Hunt Group Phone Services”, on page 621 is a “Attendant Answering Position”. The attendant answering position (AAP) feature provides support for a SIP-based AAP using a DFT, keyset telephone, or a soft client. The AAP functionality includes night service (automatic and manual control) to route calls to predefined night stations or other answering device- for example, to voice messaging, to an automated attendant application, or to a night bell device.

This is accomplished by the following provisioning steps.

1. BGL_A is created using the *CreateSubscriber* SOAP request.
2. One or more additional BGLs, BGL_B, BGL_C, and so on, are created.
3. The Simultaneous Ringing (SRS) feature is assigned to BGL_A using the *UpdateSubscriberFeatures* SOAP request.
4. The Attendant Answering Position feature is assigned to the BG, providing a Night Bell CPU ID.
5. BGL_A, BGL_B, BGL_C, etc are all added to the Night Bell CPU Group using the *UpdateSubscriberBGInfo* SOAP request.

The result is a group of lines, all of which ring simultaneously and are members of the same CPU Group. When a Hunt Group is configured with the DN of BGL_A as the Night Service DN and Night Service is active, incoming calls are diverted to BGL_A and all DNs in the Night Bell CPU Group ring. Any BGL in the BG may then dial a new Night Bell CPU access code to pick up the incoming call. In this scenario, the service logic for the Night Service feature must determine the Night Bell CPU ID for the BG. The SOAP Server allows the provisioning BgNightBellCPU service with the attribute described in the following table.

Table 416 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Night Bell CPU ID	The CPU ID of the Night Bell CPU Group. The SOAP Server verifies that the specified value is within the valid range. Note that the provided CPU ID may represent the ID of an already existing CPU Group – the administrator should be aware of this. This feature is only applicable at the BG level, and the data associated with this feature is not propagated to the BGLs.	int	1..10000	N/A

Table 416 CfBgNBCPU – Night Bell CPU Input and Output Parameters

10.4 Authorization Code (CfAuthCode)

The Business Group Authorization Codes feature provides the capability to control access to calls to parties outside the business group. The system administrator specifies the number of digits of the account code. It can be from 2 to 14 digits long; its length is the same for all stations in a business group.

To access this feature, the subscriber dials the public network access code (usually 9), followed by the destination digits. The system then prompts the subscriber to enter the authorization code. After the subscriber does so, one of the following events takes place:

- **If the authorization code is valid**: The call completes normally.
- **If the authorization code is invalid**: The system prompts the subscriber to re-enter the authorization code. If the second entry is also invalid, the call is given intercept treatment.

This service is available only for BGLs and not for residential subscribers.

Table 417 lists the *input and output parameters* associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
Enabled	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False
CORName	Class Of Restrictions Name	32	(1…32)	N/A
AuthCodeCDR	Specifies how Authorization Code is to be saved in CDR records. It may be saved as an authorization code, as an account code, as both authorization code and account code, or may not be saved at all.	Enum	NoAuthAcctCode AuthCode AcctCode BothAuthAcctCode	NoAuthAcct Code
OffnetAuthCode	Specifies which traffic types require Off-Net Authorization.	Structure		False

Table 417 CfAuthCode – Authorization Code Input and Output Parameters

10.5 Call Transfer for Endpoint (SIP Applications) (CfEpCT)

SIP Endpoints, configured with the endpoint profile, were restricted from invoking the call transfer feature. As an alternative, such endpoints employed the re-invite mechanism with the result that two channels were held up for the duration of the call. In an IVR or voice mail system, such a mechanism had a huge impact on performance and capacity. The call transfer mechanism is needed for Genesys and other IVRs and Xpressions. In the case of Xpressions, if the call transfer

feature functionality is available, all SIP ports on XPressions could use a single SIP connection for the calls. Further, this single SIP connection could be secured by TLS. The CfEpCT feature is applicable only to endpoint profiles.

Table 418 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
Enabled	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False
Comments	CfEpCT feature was introduced in WSDL version 12.00.01.23			

Table 418 Call Transfer for Endpoint (SIP Applications) Input and Output Parameters

10.6 Completion of Calls on Busy Subscriber (CfCCBS)

The Completion of Calls on Busy Subscriber service (CCBS) enables calling subscriber A to be recalled when called subscriber B becomes free. When subscriber A accepts the CCBS recall, a call is re-initiated to destination B. The CCBS is introduced as a new service to meet the overall requirement, even though its functionality is similar to the existing features Automatic Callback (AC) and Automatic Recall (AR). The CCBS service can be provisioned to all subscribers on a switch-wide basis or to individual subscriber per subscription basis. When the CCBS service is subscribed, the generic features Completion of Calls (CCS) and Deactivate Completion of Calls (DCCS) are also automatically assigned to the subscriber. Likewise, the removal of both CCBS and CCNR unassigns CCS and DCCS from the subscriber automatically.

Table 419 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required

Table 419 CfCCBS – Completion of Calls on Busy Subscriber Input and Output Parameters

10.7 Completion of Calls on No Reply (CfCCNR)

The Completion of Calls on No Reply service (CCNR) enables calling subscriber A to be notified when called subscriber B does not answer and has became free after having initiated an activity. When subscriber A accepts the CCNR recall, a call is re-initiated to destination B. The CCNR is introduced as a new service to meet the overall requirement, even though its functionality is similar to the existing feature Automatic Callback (AC). The CCNR service can be provisioned to all subscribers on a switch-wide basis, or to individual subscriber per

subscription basis. When the CCNR service is subscribed, the generic features CCS and DCCS are automatically assigned to the subscriber. Likewise, the removal of the CCNR service unassigns CCS from the subscriber automatically.

Table 420 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4.2, “Field Definitions”, on page 67.			required

Table 420 Completion of Calls on No Reply Parameters

10.8 Call Forward Invalid (CfCFID)

The CFID feature is an endpoint profile feature that will allow the administrator to select an alternate target. The specified alternate target will be any dialable DN (number) and should successfully translate into valid destinations routable via the OpenScape Voice system. If the translation of the alternate target fails, the legacy capability (routing to media server intercept) will be retained. If no alternate target is specified, the service will remain disabled, although assigned to the Endpoint Profile.

Table 421 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
Enabled	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False
ServiceId	Alternate Target (DialableDN) that should successfully translate into valid destinations routable via OpenScape Voice. This field is mandatory in case of Enabled=true.	char array	1..30	N/A

Table 421 Call Forward Invalid Information Parameters

10.9 Call Forward System Internal External (CfCFSIE)

This feature allows the subscriber to configure different forwarding destinations to be used for calls that are internal to the subscriber’s Business Group and calls that are external to the Business Group. Separate destinations can be configured for the forwarding of all calls, forwarding of calls when the subscriber is busy, forwarding of calls when the subscriber does not answer, and forwarding of calls when the subscriber has “do not disturb” active.

These scenarios correspond to those scenarios that would be handled by the Call Forwarding - All Calls (CFV), Call Forwarding Busy Line (CFB), Call Forwarding Don't Answer (CFDA), and Do Not Disturb (DND) services respectively, but note that the CFSIE feature is separate and distinct from those services, and the data related to CFV, CFB, CFDA, and DND, including the Enabled state of the feature, is configured as part of the CFSIE feature, and not by assignment of those services separately.

Also note, however, that the Voice Mail (VM) service does interact with this service, in that if the destination to which the DN is forwarded in the CFSIE service and the subscriber desires the message waiting indications to be enabled (VMWI and AMWI), the VM feature must be subscribed separately and those attributes set accordingly. The VM feature does not need to be active, nor is the destination associated with VM used by CFSIE.

This feature is allowed only for Business Group Lines and does not apply to residential subscribers.

Table 422 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.			required
Enabled	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.			False
DetermineCallBasedOnTransferringNum		boolean	True / False	
RingDurationIntCall	Indicates the duration for which the phone will ring before (if not answered) the call will be redirected. (Internal calls).Used in connection with the "no answer" portion of this service feature.	integer	0..60	24 seconds
RingDurationExtCall	Indicates the duration for which the phone will ring before (if not answered) the call will be redirected. (external calls). Used in connection with the "no answer" portion of this service feature.	integer	0..60	24 seconds
CFAllInternalDest	The number to which internal calls are forwarded when Enabled is true and CFAllEnabled is true.	Please refer to the definition of Forwarding Destination in Section 5.4, "Primitive Data Definitions", on page 66.		required if the feature is Enabled
CFAllExternalDest	The number to which external calls are forwarded when Enabled is true and CFAllEnabled is true.	Please refer to the definition of Forwarding Destination in Section 5.4, "Primitive Data Definitions", on page 66.		required if the feature is Enabled
CfAllEnabled	Indicates whether the forwarding of all calls is Enabled.	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.		False

Table 422 CfCFSIE – Call Forward System Internal External Input and Output Parameters (Sheet 1 of 3)

Field Name	Description	Length	Range/Values	Default Value
CFAllEnabledExternal	Indicates whether the forwarding of external calls is Enabled.	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66		False
CFAllPrivilegedInternal	Indicates whether phones are privileged to set CfAllEnabledinternal.			
CFAllPrivilegedIExternal	Indicates whether phones are privileged to set CfAllEnabledExternal			
CFBusyInternalDest	The number to which internal calls are forwarded when Enabled is true and CFBusyEnabled is true and the subscriber's phone is busy.	Please refer to the definition of Forwarding Destination in Section 5.4, “Primitive Data Definitions”, on page 66.		At least one destination (either internal or external) is required if the feature is Enabled.
CFBusyExternalDest	The number to which external calls are forwarded when Enabled is true and CFBusyEnabled is true and the subscriber's phone is busy.	Please refer to the definition of Forwarding Destination in Section 5.4, “Primitive Data Definitions”, on page 66.		
CFBusyEnabled	Indicates whether the forwarding of busy calls is Enabled.	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		False
CFBusyEnabledExternal	Indicates whether the forwarding of busy external calls is Enabled.			
CFBusyPriviledgedInternal	Indicates whether phones are privileged to set CFBusyEnabled.			
CFBusyPriviledged External	Indicates whether phones are privileged to set CFBusyEnabledExternal			
CFNoAnswerInternalDest	The number to which internal calls are forwarded when Enabled is true and CFNoAnswerEnabled is true and the subscriber does not answer.	Please refer to the definition of Forwarding Destination in Section 5.4, “Primitive Data Definitions”, on page 66.		At least one destination (either internal or external) is required if the feature is Enabled.
CFNoAnswerExternalDest	The number to which external calls are forwarded when Enabled is true and CFNoAnswerEnabled is true and the subscriber does not answer.	Please refer to the definition of Forwarding Destination in Section 5.4, “Primitive Data Definitions”, on page 66.		
CFNoAnswerEnabled	Indicates whether the forwarding of not answered calls is Enabled.	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		False
CFNoAnswerEnabledExternal	Indicates whether the forwarding of not answered external calls is Enabled.			
CFNoAnswerPrivilegedInternal	Indicates whether phones are privileged to set CFNoAnswerEnabled			
CFNoAnswerPrivilegedExternal	Indicates whether phones are privileged to set CFNoAnswerEnabledExternal			
CFDNDInternalDest	The number to which internal calls are forwarded when Enabled is true and CFDNDEnabled is true and the subscriber is in the “do not disturb” state.	Please refer to the definition of Forwarding Destination in Section 5.4, “Primitive Data Definitions”, on page 66.		At least one destination (either internal or external) is required if the feature is Enabled.
CFDNDExternalDest	The number to which external calls are forwarded when Enabled is true and CFDNDEnabled is true and the subscriber is in the “do not disturb” state.	Please refer to the definition of Forwarding Destination in Section 5.4, “Primitive Data Definitions”, on page 66.		

Table 422 CfCFSIE – Call Forward System Internal External Input and Output Parameters (Sheet 2 of 3)

Field Name	Description	Length	Range/Values	Default Value
CFDNDEnabled	Indicates whether the “do not disturb” state is Enabled and calls are to be forwarded.	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		False

Table 422 CfCFSIE – Call Forward System Internal External Input and Output Parameters (Sheet 3 of 3)

10.10 Call Forward Unreachable[1] (CfCFD)

This feature provides call forwarding capability to take place when a subscriber’s phone is unreachable due to being unregistered, not responding, or audit blocked. Note that this is not absolute, as there are features with higher priorities that supersede this forwarding.

This service applies to SIP subscribers, and SIP keysets, both primary and phantom keyset lines. BGLs can subscribe to the feature, and it can also be assigned to feature profiles.

Table 423 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
Enabled	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False
ServiceId	Indicates the destination number. Can be specified only if subscribed to; must be specified when this feature is subscribed to and enabled. Can be up to 30 digits in length. Is validated against any existing TRS Code restrictions.	Please refer to the definition of Forwarding Destination in Section 5.4, “Primitive Data Definitions”, on page 66.		required if the feature is Enabled
ExternalServiceID	Indicates an external destination number that can be specified in the request. Please note that the service ID specified here should be a routable service ID.	Please refer to the definition of Forwarding Destination in Section 5.4, “Primitive Data Definitions”, on page 66.		Not Mandatory

Table 423 Call Forward unreachable Information Parameters

10.11 Call Forwarding [2]- All Calls (CfCFV)

The OpenScape Voice system-based Call Forwarding - All Calls feature provides the capability to redirect calls intended for the subscriber to another destination. The subscriber activates and deactivates the feature and specifies the forwarding destination.

1. Call forward unreachable is sometimes known as call forward dependable.
2. Can be referred to as Station Call Forwarding All Calls.

This feature can be provisioned at the business group level with a denied option at the subscriber level. Also known as Call Forwarding Variable or Call Forward Unconditional. This feature overrides CfCFB and CfCFDA.

Table 424 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
Enabled	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False
ServiceId	Indicates the destination number. Can be specified only if subscribed to; must be specified when this feature is subscribed to and enabled. Can be up to 30 digits in length. Is validated against any existing TRS Code restrictions.	Please refer to the definition of Forwarding Destination in Section 5.4, “Primitive Data Definitions”, on page 66.		required if the feature is Enabled
SubActivatable	Indicates the method by which the subscriber can activate/deactivate this feature. May be one of the following values:	enum	• None — Subscriber cannot activate or deactivate this feature • Phone — The subscriber can activate or deactivate this feature via the Phone • Web — The subscriber can activate or deactivate this feature via the Web. • All — The subscriber can activate or deactivate this feature via either the Web or phone interfaces.	All

Table 424 CfCFV – Call Forward Unconditional Input and Output Parameters (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
DestServiceId SubControllable	Indicates the method by which the subscriber can change the destination of Call Forwarding. This field can be set to Phone only if SubActivatable is set to Phone or All since the subscriber can change only the destination when the feature is activated. May be one of the following values.	enum	• None — Subscriber may not change the destination for this feature. • Phone — The subscriber may change the destination for this feature via the Phone • Web — The subscriber may change the destination for this feature via the Web. • All — The subscriber may change the destination for this feature via either the Web or Phone interfaces.	None
CFNotifyCallingPart y	If set to CfNotifyCallingPtyNone, no announcement is played to the calling party. If set to CfNotifyCallingPtyWithFwdNumber, an announcement with the forwarding number is played to the calling party. If set to CfNotifyCallingPtyNoFwdNumber, the announcement played to the calling party does not include the forwarding number. There are three values for the parameter: CfNotifyCallingPtyNone CfNotifyCallingPtyWithFwdNumber CfNotifyCallingPtyNoFwdNumber	enum		CfNotify CallingPty None
NotifySubCallFwdO ccurred	If this parameter is set to True, a splash ring will notify the called party that their call is being forwarded. The default value is False.	Boolean	True, False	False
NotifySubCallFwdA ctive	If set to True, the user is reminded that forwarding is currently active. The notification method can be either tone or announcement, which is part of the treatment provisioning, not part of the feature provisioning.	Boolean	True, False	False

Table 424 CfCFV – Call Forward Unconditional Input and Output Parameters (Sheet 2 of 2)

10.12 Call Forwarding Busy Line (CfCFB) [1]

The OpenScape Voice system-based Call Forwarding Busy Line (CFBL) feature provides the capability to redirect calls intended for the subscriber to another destination when the subscriber's station is in use. The subscriber activates and deactivates the feature, and specifies the forwarding destination.

This feature can be provisioned at the business group level with a denied option at the subscriber level.

Table 425 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.			required
Enabled	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.			False
ServiceId	Indicates the destination number. Can be specified only if subscribed to; must be specified when this feature is subscribed to and enabled. Can be up to 30 digits in length. Is validated against any existing TRS Code restrictions.	Please refer to the definition of Forwarding Destination in Section 5.4, "Primitive Data Definitions", on page 66.		required if the feature is Enabled
SubActivatable	Indicates the method by which the subscriber can activate/deactivate this feature. May be one of the following values. If not specified, will default to None.		• None — subscriber cannot activate or deactivate this feature. • Phone — the subscriber can activate or deactivate this feature via the phone. • Web — the subscriber can activate or deactivate this feature via the Web. • All — the subscriber can activate or deactivate this feature via either the Web or phone interfaces.	None.
DestServiceIdSub Controllable	Indicates the method by which the subscriber can change the destination of Call Forwarding. Can be one of the following values. This field may only be set to Phone only if SubActivatable is set to Phone or All as the subscriber can only change the destination when the feature is activated.		• None — Subscriber may not change the destination for this feature. Default. • Phone — The subscriber may change the destination for this feature via the Phone • Web — The subscriber may change the destination for this feature via the Web. • All — The subscriber may change the destination for this feature via either the Web or phone interfaces.	None

Table 425 CfCFB – Call Forward Busy Input and Output Parameters (Sheet 1 of 2)

1. Can be referred to as Station Call Forwarding Busy Line.

Field Name	Description	Length	Range/Values	Default Value
CFNotifyCalling Party	There are 3 values for the parameter: • CfNotifyCallingPtyNone • CfNotifyCallingPtyWithFwd Number • CfNotifyCallingPtyNoFwd Number If set to CfNotifyCallingPtyNone, no announcement is played to the calling party. If set to CfNotifyCallingPtyWithFwd Number, an announcement with the forwarding number is played to the calling party. If set to CfNotifyCallingPtyNo FwdNumber, the announcement that is played to the calling party does not include the forwarding number.			CfNotifyCallingPt yNone
NotifySubCallFwdActi ve	If set to True, the served user is reminded that call forwarding is currently active. The notification method can be either tone or announcement, which is part of the treatment provisioning, not part of the feature provisioning.		True, False	False

Table 425 CfCFB – Call Forward Busy Input and Output Parameters (Sheet 2 of 2)

10.13 Call Forwarding Don't Answer (CfCFDA) [1]

The OpenScape Voice system-based Call Forwarding Don't Answer (CFDA) feature provides the capability to redirect calls intended for the subscriber to another destination if the call is not answered after a preset number of rings. The subscriber activates and deactivates the feature and specifies the forwarding destination.

This feature can be provisioned at the business group level with a denied option at the subscriber level.

Table 426 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.			required
Enabled	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.			False

Table 426 CfCFDA – Call Forwarding Don't Answer Input and Output Parameters (Sheet 1 of 3)

1. Can be referred to as Station Call Forwarding Don't Answer.

Field Name	Description	Length	Range/Values	Default Value
ServiceId	Indicates the destination number. Can be specified only if subscribed to; must be specified when this feature is subscribed to and enabled. Can be up to 30 digits in length. Is validated against any existing TRS Code restrictions.	Please refer to the definition of Forwarding Destination in Section 5.4, "Primitive Data Definitions", on page 66.		required if the feature is Enabled
NumberOfRings	This parameter is obsolete and has been replaced with RingDuration, which indicates the number of ring cycles after which the call will be redirected. Assumes six-second ring cycles at phone.		1 – 10	4 (24 seconds)
RingDuration	Indicates the duration for which the phone will ring before (if not answered) the call will be redirected.		0-60	24 seconds
SubActivatable	Indicates the method by which the subscriber can activate/deactivate this feature. May be one of the following values:		• None — Subscriber cannot activate or deactivate this feature • Phone — The subscriber can activate or deactivate this feature via the Phone • Web — The subscriber can activate or deactivate this feature via the Web. • All — The subscriber can activate or deactivate this feature via either the Web or phone interfaces.	None

Table 426 CfCFDA – Call Forwarding Don't Answer Input and Output Parameters (Sheet 2 of 3)

Field Name	Description	Length	Range/Values	Default Value
DestServiceId SubControllable	Indicates the method by which the subscriber can change the destination of Call Forwarding. This field can be set to Phone only if SubActivatable is set to Phone or All since the subscriber can change only the destination when the feature is activated. May be one of the following values.		• None — Subscriber may not change the destination for this feature. • Phone — The subscriber may change the destination for this feature via the Phone • Web — The subscriber may change the destination for this feature via the Web. • All — The subscriber may change the destination for this feature via either the Web or Phone interfaces.	None
CFNotifyCallingPart y	If set to CfNotifyCallingPtyNone, no announcement is played to the calling party. If set to CfNotifyCallingPtyWithFwdNumber, an announcement with the forwarding number is played to the calling party. If set to CfNotifyCallingPtyNoFwdNumber, the announcement played to the calling party does not include the forwarding number. There are three values for the parameter: CfNotifyCallingPtyNone CfNotifyCallingPtyWithFwdNumber CfNotifyCallingPtyNoFwdNumber			CfNotify CallingPty None
NotifySubCall FwdActive	If set to True, the user is reminded that forwarding is currently active. The notification method can be either tone or announcement, which is part of the treatment provisioning, not part of the feature provisioning.		True, False	False

Table 426 CfCFDA – Call Forwarding Don't Answer Input and Output Parameters (Sheet 3 of 3)

10.14 Call Forwarding Voice Mail (CfVM) [1]

This service indicates that the subscriber has Voice Mail. VoiceMail is a combination of CfCFDA and CfCFB. These services are mutually exclusive and an attempt to add VM or either CFDA or CFB when the other is provisioned will result in an error. The removal of CfCFDA or CfCFB will not effect CfVM and vice versa.

1. Can be referred to as Station Call Forwarding Voice Mail.

Table 427 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
Enabled	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False
NumberOfRings	This parameter is obsolete and has been replaced with RingDuration, which Indicates the number of ring cycles after which (if not answered) the call will be redirected to the Voice Mail server. Assumes that the device has six-second ring cycles.	integer	1 – 10	4 (24 seconds)
RingDuration	Indicates the duration that the phone will ring before (if not answered) the call will be redirected to the Voice Mail server.	integer	0-60	24 seconds
ServiceId	Indicates the destination number. Can be specified only if subscribed to; must be specified when this feature is subscribed to and enabled. Is validated against any existing TRS Code restrictions.	Please refer to definition of Forwarding Destination in Section 5.4, “Primitive Data Definitions”, on page 66.		N/A
VMWI	Indicates the visual message waiting indication capability (lamp capability). This is applicable only to MGCP subscribers. Not for OpenScape Voice.	Boolean	False (0) — no VMWI capability True (1) — has VMWI capability	The default value is based on a RTP parameter Srx/Main/VMWI capability Default. The default for the parameter is True.
AMWI	Indicates whether the audible message waiting indication (stutter dial tone) is enabled or disabled. This is applicable only to MGCP subscribers. Not for OpenScape Voice.	Boolean	False (0) — AMWI disabled True (1) — AMWI enabled	True

Table 427 CfVM – Voice Mail Input and Output Parameters

10.15 Call Pickup (CfBgCPU)

The Group Call Pickup feature permits stations to be combined into *pickup groups*. Pickup groups permit a member to answer a call on behalf of another member of the group.

Table 428 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
Enabled	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False

Table 428 CfBgCPU – Call Pickup Input and Output Parameters

10.16 Call Pickup Directed (CfCPUD)

This feature allows a station user to answer a call on another telephone from their telephone. The feature is allowed only for Business Group Lines and Feature Profile level (FP). The feature is applicable only to SIP subscribers with an OpenStage SIP device. Only OpenStage phones support this feature.

Additionally, only calls to BGLs within the same BG can be picked up, and only a single dialog can be picked up. Call pickup directed can be used to pick up calls in the following states:

- Ringing
- Camped on (Call waiting)

Upon subscription, CPUD shall be considered inactive. A field named “Enabled” shows that the service is inactive for a subscriber when its value is “false” and active when its value is “true”. The default will be “false” (considered inactive), when CPUD is subscribed to a BGL.

Table 429 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
Enabled	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False

Table 429 Call Pickup Directed Parameters

10.17 Called Party Name Delivery (CfCISNAME)

Obsolete

10.18 Called Party Number Delivery (CfCISNUM)

Obsolete

10.19 Calling Identity Delivery and Suppression (CfCIDS)

The Calling Identity Delivery and Suppression (CIDS) feature provides subscribers the capability to deliver or suppress their calling identity parameters (name and number).

This feature can be provisioned at the business group level with a denied option at the subscriber level. Counters are provided for feature activation attempts. Also known as Outgoing CID Suppression.

Table 430 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required

Table 430 CfCIDS – Outgoing Caller ID Suppression Input and Output Parameters

10.20 Calling Name Delivery (CfCNAM)

since V5.

10.21 Calling Name Delivery (BG) (CfBgCNAM)

Obsolete since V5.

10.22 Calling Name Delivery Blocking (CfCNAB)

Obsolete since V5.

10.23 Calling Number Delivery (CfCND)

Obsolete since V5.

10.24 Calling Number Delivery Blocking (CfCNDB)

Obsolete since V5.

10.25 CSTA for ComAssistant (CfCSTA)

This feature indicates the ability of the subscriber to use third-party call control using the Computer Supported Telephony Application, ComAssistant. A subscriber can place calls and receive them using the ComAssistant application. This feature can be subscribed to by residential subscribers or individual BGLs.

Attention: Whenever the CSTA service is removed from or denied to a subscriber or BGL, a message is sent to the CSTA Signaling Manager to tell it to stop monitoring. Likewise, if a subscriber or BGL with the CSTA service assigned is deleted, the CSTASM is informed via a message.

Table 431 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
CSTAType	Indicates the capabilities of the endpoint.		May be one of the following values: CstaTypeNorma CstaTypeSiemensType1 CstaTypeMGCPResidential CstaTypeMGCPBusiness CstaTypeCstaOverSip CstaTypeSpecial2 CstaTypeSpecial3 CstaTypeSpecial4 CstaTypeSpecial5	CstaTypeNormal
Monitoring Status	Indicates the current status of a third-party application. This parameter is read-only. It is returned in the feature list of subscribers, but not of the BG. It cannot be set. It will be either Active or Inactive.			

Table 431 CfCSTA – CSTA for ComAssistant Input and Output Parameters

10.26 Customer Originated Trace (CfMCT)

The Customer-Originated Trace feature provides subscribers the capability to generate an automatic trace of the last call received. Subscribers typically use this feature in response to malicious, harassing, or nuisance calls, in order to provide a trace over time of such activity. Also known as Malicious Call Trace.

A classmark at the subscriber level is required for access to this feature.

Table 432 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
Enabled	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. Note that unlike some other features, the default value for Enabled is true, not false.	boolean	true false	true
Billing	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			FlatRate

Table 432 CfMCT – Malicious Call Trace (Customer Originated Trace) Input and Output Parameters

10.27 Deny Terminating Feature (CfDTF)

The Deny Terminating Feature (CfDTF) enables the administrator to enable or disable Callback Monitoring from a specific subscriber.

Attention: In the future more features will be supported. Currently, there are three attributes (every new feature that can be denied will become a new attribute).

Table 432 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required

Table 433 CfDTF – Deny Terminating Feature Input and Output Parameters (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
Enabled	Please refer to Section 5.4, "Primitive Data Definitions", on page 66. Note that unlike some other features, the default value for Enabled is true, not false.	boolean	true false	false
Callback Monitoring	This attribute, is a Boolean attribute that if set TRUE, it shall indicate that callback monitoring of this subscriber as a result of the CCBS/CCNR features invoked by another subscriber is not allowed.			false

Table 433 CfDTF – Deny Terminating Feature Input and Output Parameters (Sheet 2 of 2)

10.28 E9-1-1 LIN Administration (CfBgE911LIN)

This feature may be provisioned for a Business Group and is applicable to the entire business group. It may not be provisioned against individual subscribers or Business Group Lines and the feature-related data is not propagated to the subscriber/BGL level.

E9-1-1 LIN Administration indicates that the E9-1-1 service is to be used to accomplish the routing of emergency calls from BGLs that are members of this Business Group. Also, the assignment of this service to a BG is expected to be done only by the authorized overall BG Controller for the system. The provisioning of this service to a BG indicates that the BG Administrator for that BG has the capability to use the BG Subnet commands (see Section 11.14, "Business Group Subnets", on page 447).

Table 434 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Subscribe to the service. See Section 5.4, "Primitive Data Definitions", on page 66.			required

Table 434 CfBgE911LIN – E9-1-1 LIN Administration Input and Output Parameters (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
E911 Default Emergency Number	If an emergency call cannot be routed to a PSAP operator, the call is routed to this number if it exists. This field is optional when assigning the E9-1-1 LIN Administration feature via the *UpdateBgFeatures* request. If it is supplied, the SOAP Server verifies through XLA that the number is routable, and also verifies that it corresponds to a BGL that is a member of the BG being updated. If these checks fail, the request is rejected. Note that the BGL whose DN is specified as the E9-1-1 Default Emergency Number may not be deleted unless this service is updated to remove the specification of that DN.	Please refer to definition of ServiceId in Section 5.4, "Primitive Data Definitions", on page 66. Must be the DN of an existing BGL.		""
E911 Intercept ID	In case an emergency call cannot be routed to the BG default emergency number, the call is routed to this intercept. This field is mandatory when assigning the E9-1-1 LIN Administration feature via the *UpdateBgFeatures* request. The SOAP Server verifies that the specified intercept exists in the intercept table (administered via the OpenScape Voice Assistant); if it does not, the request is rejected.	15	Character string; must represent a valid intercept.[1]	required
[1]The list of valid existing intercepts is available using the *GetInterceptList* SOAP request (see Section 7.8, "Get Intercept List").				

Table 434 CfBgE911LIN – E9-1-1 LIN Administration Input and Output Parameters (Sheet 2 of 2)

10.29 Executive Override (CfEXO)

This feature may be used to allow a subscriber to barge in to a call in progress, to prevent other subscribers from using the Executive Override feature to barge into their calls, or both. The feature is only allowed for Business Group Lines, not for residential subscribers.

Table 454 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
Enabled	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False
EXOOption	This parameter specifies whether the subscriber shall be able to use the feature to barge in, to block barge in, or both.	enum	EXO_OvrRide EXO_BlockOvrRide EXO_OvrRideAndBlockOvrRide	EXO_OvrRide

Table 435 CfECF – Enhanced Call Forward Input and Output Parameters

10.30 Distinctive Ringing (CfBgDR)

The Distinctive Ringing feature may be provisioned for a Business Group and is applicable to the entire business group. It may not be provisioned against individual subscribers or Business Group Lines and the feature-related data is not propagated to the subscriber/BGL level.

This feature provides the ability for the optiPoint 410 S and optiPoint 420 S user to hear different ringing indications for internal and external calls.

The internal ringing pattern is used for calls received from users within the business group. The external ringing pattern is used for calls received from parties outside the business group. The actual ringing cadences for a given type are defined in the telephone.

The business group administrator controls internal and external ringing for the entire business group. If this capability is not provisioned, the ringing configured locally at the telephone is provided.

Table 436 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
Enabled	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False
Comments:	The feature is added from the WSDL version 10.00.03.01.			

Table 436 CfBgDR – BG Distinctive Ringing Input and Output Parameters

10.31 Do Not Disturb (CfDND)

Do Not Disturb (DND) provides a system-based feature that allows a subscriber, who does not want to be disturbed with calls terminating to them, to have the terminating calls immediately diverted to an announcement indicating the reason for not completing the call to the expected destination. The diversion of the terminating call occurs regardless of the busy/idle status of the DND subscriber. The subscriber with this feature activates the feature to prevent the terminations and deactivates it to allow calls to terminate.

The diversion can be modified to go to other destinations (i.e., not just to an intercept) based on the setting of the RTP parameter Srx/Main/CFSIECompatibility.

Originating calls are still possible when DND is active.

Note the following interactions between the DND feature and the CFSIE feature in Section 10.9, “Call Forward System Internal External (CfCFSIE)”:

- DND and CFSIE-DND can be assigned to the same subscriber.
- If DND is not active, then the status of CFSIE-DND is not checked in the processing of a call.
- If DND is active, then routing occurs based on the setting of the RTP parameter Srx/Main/CFSIECompatibility,
  - When set to Yes,
    - If CFSIE-DND is assigned and active, and the type of call (internal/external) has a forward-to DN provisioned, then the call is routed to that forward-to DN.
    - Otherwise if CFDA is assigned and active, then the call is routed to that forward-to DN.
    - Otherwise if CFVM is assigned and active, then the call is routed to that forward-to DN.

      Note: CFVM cannot be active at the same time as any of the 4 CFSIE features.

    - Otherwise if CFSIE-DA is assigned and active, and the type of call (internal/external) has a forward-to DN provisioned, then the call is routed to that forward-to DN.
    - Otherwise, DND intercept treatment will occur.
  - When set to No, DND Intercept Treatment will occur.

When a call is forwarded to a DN with DND active, normal feature interaction processing occurs, as mentioned above.

Table 437 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
Enabled	Indicates whether the feature is active or inactive. If not specified initially, defaults to False (inactive). Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False

Table 437 CfDND - Do Not Disturb Input and Output Parameters

10.32 Hot Desking (CfHotDesk)

The Hot Desking feature provides subscribers the capability to log on to and use a telephone in another office. The telephone in the other office (the remote office telephone) then has all of the same OpenScape Voice system-provided features and capabilities as the telephone in the subscriber's office (the home office telephone). This feature is also known as *hoteling*.

To operate, Hot Desking must be assigned to both the home office telephone and on the remote office telephone. Telephone-based features are not transferred from the home office telephone to the remote office telephone. The remote office telephone and the home office telephone must be hosted by the same OpenScape Voice system, and must both be SIP telephones.

Table 438 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
Enabled	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False
HotDeskSide	The type of hot desking feature to be assigned to the DN.	Enum	HDRemote HDHomeBase	required
SessionStatus	This is a read-only field that displays the session status of the Hotdesking session.	Boolean	True, False	

Table 438 CfHotDesk – Hot Desking Input and Output Parameters

10.32.1 Hot Desking (BG) (CfBgHotDesk)

This feature needs to be assigned at the BG level before any DN in the BG can be assigned the Hot Desking Feature.

Table 439 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.			required
autoLogoffTime	The time at which all BGLs Hot desking session will be automatically terminated.	TOD	Hh:mm format.	
DisableAutoLogOff	Indicates whether auto logoff is enabled or disabled for hot desking subscribers in the business group.	boolean	true false	false

Table 439 CfBgHotDesk – BG Hot Desk Input and Output Parameters

10.33 Keyset Operation (CfBgKS)

This feature may be provisioned for a Business Group and is applicable to the entire business group. It may not be provisioned against individual subscribers or Business Group Lines and the feature-related data is not propagated to the subscriber/BGL level.

Keyset Operation indicates that the Business Group Lines (BGLs) that are members of this Business Group may be configured as keyset devices and use keyset operations, which encompass the following capabilities:

- SIP telephones may be provisioned with multiple Directory Numbers (DNs) and a DN may be provisioned on multiple SIP telephones.
- Calls may be originated from any DN that appears on the telephone. Calls to a DN are signaled simultaneously at all telephones where the DN appears and can be answered from any of those telephones.
- The appearances on a telephone are typically associated with line keys, which have LEDs to show the status of the line (busy, alerting, holding).

- A call can be placed on hold at one telephone and retrieved from hold at another telephone that has an appearance of the DN.

Note: The SOAP Server does not restrict the provisioning of keyset information to only those BGLs that are members of a BG that has this service subscribed. A SOAP client, however, could do so. For further information about provisioning keyset information to a sub/BGL, refer to Section 4.13, “Keyset Information”, on page 55.

Table 440 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
ManualHold RecallTimer	The time during which recall is performed after the expiration of the BGL-level ManualHoldTimer. After the timer expires, the call can be sent to intercept.	int	5..3600	18
Line Reservation Timer	Used to unreserve the line in cases where the line needs to be cleared due to non-use.	int	10..300	120

Table 440 CfBgKS – Keyset Operation Input and Output Parameters

10.34 Station Controlled Conference[1] (CfBgLC or CfSCC)

The station-controlled conference feature provides subscribers the capability to establish a conference call with up to 16 participants on an ad-hoc (on-demand) basis. Conference participants can be members of the same business group, another business group, or in the public network.

SCC ad-hoc conference can be initiated by any station with the service subscribed. Any participant in a large conference with this feature assigned is considered controller or initiator and as result can add other participants to the conference. Participants with SCC assigned are active participants and can add others to the conference. A SIP subscriber without SCC assigned can be added to the conference, but then is considered a passive participant and cannot add others to the conference.

Table 441 lists the input and output parameters associated with this feature.

1. Station Controlled Conference is also known as Large Conference.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.			required
Minimum Number of Ports	The number of reserved ports for large conference service for the BG.	int	0.. number of available conference ports not already reserved by other BGs	0
Maximum Number of Ports	The maximum number of large conference ports that the BG can use simultaneously	int	0.. number of available conference ports not already reserved by other BGs	200
Maximum Number of Conferences	Maximum number of conferences in simultaneous per BG.	int	1.. maximum number of ports / 3	1
Maximum Number of Conference Members	Maximum number of participants in a conference, defined per BG.	int	3.. 16 or maximum number of ports if minor than 16	3
Conference Recall Disconnect Timer	Timer that a recall to conference will ring at the device before disconnecting the call.	int	1..254	30
PlayEntryExitTone	Play tone on conference entry/ exit.	bool	true false	true
BlockTransferToRestrictedParty	Allows or disallows the check whether a subscriber based on its TRS settings is allowed to call another party, based on its Traffic type. If the option is activated and two parties remain in a conference, where one party is restricted to call the second party, the conference is forced to be released.	bool	true false	false

Table 441 CfBgLC – Station Controlled Conference Input and Output Parameters

Considerations for validation:

Minimum Number of Conference Ports <= Maximum Number of Conference Ports

Number of available conference ports = Number of System Conference Ports - summation of (Minimum Number of Conference Ports) where the summation is taken over all provisioned BGs

10.35 Music On Hold (CfMOH)

The music on hold line and group options feature provides the capability for subscribers to hear music when they are placed on hold. The feature can be assigned to an individual user, an entire business group, or an entire hunt group. When provisioned, music is provided for all instances when a subscriber places

a caller on hold. This includes consultation hold, call hold, manual hold, CSTA-initiated hold, and queuing. The source of the music is be specified by an intercept name. As with any intercept name, this is provisioned via the OpenScape Voice Assistant/CLI.

Table 442 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Intercept Name	Identifies the intercept to be used to provide the music source. The SOAP Server verifies that the specified intercept exists in the intercept table (administered via the OpenScape Voice Assistant); if it does not, the request is rejected. This field is optional and, if not specified, the *MOH_Default_Tr* intercept is used. The default intercept cannot be administered by the OpenScape Voice Assistant/CLI operator, but is created by the MOH service at startup. Note that although the interface includes the intercept name as a character string, the internal storage of this data is the intercept ID, which is a numeric value. Therefore, the SOAP Server reads the `INTERCEPT_T` table, given the intercept name provided, to determine the corresponding intercept ID, which is stored in the `SERVICE_DATA_T` table.	15	Character string – must represent a valid intercept.[1]	MOH_Default_Tr
[1]The list of valid existing intercepts is available using the *GetInterceptList* SOAP request (see Section 7.8, "Get Intercept List").				

Table 442 CfMOH – Music On Hold Input and Output Parameters

10.36 One Number Service (CfONS)

The One Number Service via Associated Devices (CfONS) provides the service for the subscriber to originate and receive calls using an associated device other than the subscriber's own DN. The associated devices could be the home phone, cell phone, and so on, and the devices could be resided outside of the OpenScape Voice platform.

ONS Outbound allows calls to be originated from an associated device, and ONS Inbound allows calls to be received at an associated device. For ONS Inbound, the incoming call is offered to an application for routing instructions, and the application can accept the call at the subscriber's registered device, or specify an associated device to receive the call. In either variant, only the subscriber DN is made known to the other party regardless of whether any of the associated devices are in use.

This service is applies only to SIP subscribers. It is required to have the CSTA service before it can assign the One Number Service to the subscriber.

Table 443 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.			required
Enabled	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.			False
ONSType	The type of ONS service to be provided. OutboundOnly enables a subscriber to originate a call via a CSTA application user interface through an associated device (such as a mobile phone or hotel phone). InboundAndOutbound allows both the above functionality and inbound functionality, which enables a subscriber to receive and answer a call at an associated device (such as a mobile phone or home phone) when the subscriber DN is called. The ONS subscriber must be a SIP device. Including non-registering subscriber support for all new CSTA services requires SOAP to not restrict ONS-IO provisioning on provisioned-only DNs.	enum	OutboundOnly InboundAndOutbound	OutboundOnly
StaticOND	V7.00.18.01 upwards			
Enabled	Indication of the status of the Call Forwarding – Static OND feature	Boolean	True False	False
ServiceId	The preferred device target for ONS binding on inbound calls. By default its value should be left blank, meaning that ONS = OND	No	See definition of ServiceId in Section 5.4, "Primitive Data Definitions", on page 65.	

Table 443 CfONS – One Number Service Input and Output Parameters

10.37 Remote Activation of Call Forwarding (CfRACF) [1]

The Remote Activation Call Forwarding (RACF) feature is an optional capability of the Call Forwarding - All Calls feature. It provides the subscriber the capability to activate, deactivate, and change the redirect number for call forwarding-all

1. Can be referred to as Station Call Forwarding Remote Activation.

calls from a locations other than the subscriber's station. This capability permits the subscriber to manage call forwarding options and change forwarding destinations from home or from another work location.

To subscribe to this feature, the subscriber must also have the call forwarding-all calls feature.

Table 444 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
Pin	The Personal Identification Number (PIN) that the subscriber must enter for authorization. If not specified, the service uses the PIN from the subscriber profile. If this PIN is specified, it overrides the PIN in the profile.	8	Character string	“”

Table 444 CfRACF – Remote Activation of Call Forward Variable Input and Output Parameters

10.38 Remote Call Forwarding (CfRCF) [1]

The OpenScape Voice system-based Remote Call Forwarding (RCF) feature provides the capability to redirect calls intended for subscriber to a fixed destination. RCF cannot be enabled or disabled by subsequent SOAP requests; it is always on. If RCF is provisioned, no other service feature can be assigned to the subscriber. The calling party receives no indication that the call terminates at a remote location.

This feature is similar to Call Forwarding - All Calls, with the following exceptions:

- RCF can only be assigned to a profile-only subscriber: there is no actual phone associated with this DN and the subscriber cannot make calls from this number.
- A specified number of simultaneous calls can be forwarded to the remote station from the RCF base DN.

Table 445 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required

Table 445 CfRCF – Remote Call Forward Input and Output Parameters (Sheet 1 of 2)

1. Can be referred to as Station Remote Call Forwarding.

Field Name	Description	Length	Range/Values	Default Value
ServiceId	The number to which all incoming calls are to be routed.	See definition of Forwarding Destination in Section 5.4, “Primitive Data Definitions”, on page 66.		required
Max Simultaneous Calls	The maximum number of calls allowed to be forward at any one time. This field can be a maximum of 32.	integer	0 to 2500	32

Table 445 CfRCF – Remote Call Forward Input and Output Parameters (Sheet 2 of 2)

10.39 Selective Call Acceptance (CfSCA)

The Selective Call Acceptance feature provides the capability to build a list of numbers (known as a *screening list*) from which the subscriber wants to accept incoming calls. Callers who are not on the screening list hear a denial announcement that they are being forwarded to another number or being played a denial announcement.

This feature can be provisioned at the business group level with a denied option at the subscriber level.

Table 446 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
Enabled	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False
Billing	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			FlatRate
ScreenList	The list of DNs from which calls are to be accepted. The entries should be fully qualified national numbering plan formatted numbers. The individual entries are not editable and the entire list must be specified.	32 entries; up to 15 digits each	List of numeric strings.	N/A
Pin	Defines the priority code that a caller can use to be placed through to the subscriber even though they are calling from a phone that is not on the screening list.	8	Character string.	N/A

Table 446 CfSCA – Selective Caller Accept Input and Output Parameters (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
Sca Termination Treatment	Indicates the denial treatment for callers who are not on the list; either announcement or forwarding is possible.	enum	ScaTerminationDenialAnnouncement, ScaTerminationToRemoteDn	Sca Termination Denial Announce-ment
ServiceId	Indicates the destination number to forward denied calls to when ScaTermination Treatment = Sca TerminationToRemoteDn. Is validated against any existing TRS Code restrictions.	See definition of Forwarding Destination in Section 5.4, “Primitive Data Definitions”, on page 66.		“”

Table 446 CfSCA – Selective Caller Accept Input and Output Parameters (Sheet 2 of 2)

10.40 Selective Call Forwarding (CfSCF) [1]

The Selective Call Forwarding feature provides the capability to build a list of numbers (a *screening list*) which the subscriber wants to automatically forward to another destination. When a caller's number matches a number on the forwarding list, the call is forwarded to the redirect number. This can be another telephone, another subscriber, voice mail, or an announcement. When the caller's number is not on the forwarding list, the call is completed as usual.

This feature can be provisioned at the business group level with a denied option at the subscriber level.

Table 447 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
Enabled	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False
ServiceId	Indicates the destination number to which calls are forward when callers are found to be in the screening list. Is validated against any existing TRS Code restrictions.	See definition of Forwarding Destination in Section 5.4, “Primitive Data Definitions”, on page 66.		required if Enabled is True
SplashRing	Indicates whether the subscriber's phone will ring once as a reminder when a call is forwarded.	Boolean	True, False	False

Table 447 CfSCF – Selective Call Forward Input and Output Parameters (Sheet 1 of 2)

1. Can be referred to as Station Selective Call Forwarding.

Field Name	Description	Length	Range/Values	Default Value
ScreenList	List of calling numbers to forward. Callers on this list will not be put through to the subscriber, but instead will be forwarded to the specified destination. The entries should be fully qualified national numbering plan formatted numbers. The individual entries cannot be edited and the entire list must be specified.	32 entries; up to 15 digits each	List of numeric strings.	N/A
BlackList	Specifies whether the ScreeningList of DNs associated with the SCF feature is a white list (only calling DNs on the list are forwarded) or a black list (all calling DNs are forwarded except those on the list). **NOTE**: This attribute is not supported for Version 2.2.	Boolean	True, False	False

Table 447 CfSCF – Selective Call Forward Input and Output Parameters (Sheet 2 of 2)

10.41 Selective Call Rejecting (CfSCR)

The Selective Call Rejection feature provides the capability to build a list of numbers (known as a *screening list*) from which the subscriber does not want to accept incoming calls. When a caller's number does not match a number on the rejection list, the call is completed. When the caller's number matches a number on the rejection list, the caller hears an announcement that indicates the subscriber does not accept calls from the number.

This feature can be provisioned at the business group level with a denied option at the subscriber level.

Table 448 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
Enabled	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False

Table 448 CfSCR – Selective Caller Rejection Input and Output Parameters (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
SubControlled	Indicates whether the subscriber can enable/disable the service or whether it is controlled by the telco only. If not specified, defaults to telco only.	Boolean	True, False	false
Billing	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			FlatRate
ScreenList	List of calling numbers to reject. Callers on this list will not be put through to the subscriber. The entries should be fully qualified national numbering plan formatted numbers. The individual entries cannot be edited and the entire list must be specified.	32 entries; up to 15 digits each	List of numeric strings.	N/A

Table 448 CfSCR – Selective Caller Rejection Input and Output Parameters (Sheet 2 of 2)

10.42 Serial Ringing (CfSERRNG)

The serial ringing feature provides subscribers the capability to be sequentially rung at a series of locations. The subscriber or the system administrator creates a serial ringing list that contains up to six DNs. These DNs represent the additional locations that ring when an incoming call arrives at the main DN, and the sequence in which they are rung.

This feature optionally includes the ability for the caller to instantly transfer to the caller's voice mailbox, rather than waiting for the call to progress through all locations to do so.

Table 449 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Subscribe to the service. Refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
Enabled	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False
Duration	Period of time the primary phone rings before the DNs in the list are dialed.	integer	1–120	18
SerialDNList	A list of up to six entries, each with the attributes listed below.			

Table 449 CfSERRNG – Serial Ringing Input and Output Parameters (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
(SerialDNList Entry) ServiceId	A DN to be rung serially.	Please refer to the definition of Forwarding Destination in Section 5.4, “Primitive Data Definitions”, on page 66.		“”
(SerialDNList Entry) DNActive	Indicates whether this DN is to be rung (True) or skipped (False).	Please refer to definition of Enabled in Section 5.4, “Primitive Data Definitions”, on page 66.		False
(SerialDNList Entry) Duration	Period of time this entry’s phone rings before the next DN in the list is dialed.	integer	1-120	18
(SerialDNList Entry) PPSState	Indicates whether the this entry’s DN is public or private.	enum	Public, Private	Public
(SerialDNList Entry) Extension	Indicates that the DN is an extension, or intercom dialing code (True). If False, the number is not an extension, but a fully qualified national numbering plan formatted number.	Boolean	True, False	False

Table 449 CfSERRNG – Serial Ringing Input and Output Parameters (Sheet 2 of 2)

10.43 Simultaneous Ringing (CfSRS)

The Simultaneous Ringing feature provides subscribers the capability to be simultaneously rung at multiple locations. The subscriber or the system administrator creates a simultaneous ringing list that contains up to six DNs. These DNs represent the additional locations that ring when an incoming call arrives at the main DN. After the feature is activated, incoming calls cause the main DN and each destination DN to ring. The first DN to answer is connected; ringing then stops on the other DNs. If the call is forwarded to another DN, such as voice mail, it rings until answered.

Remote Feature Access (RFA) was designed to activate/deactivate the SRS feature from a “remote” telephone (any DN other than the one to which the features are subscribed). RFA can only be used in conjunction with SRS; therefore, subscriber provisioning will block the assignment of RFA if SRS is not also subscribed or being subscribed in the same request.

Table 450 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required

Table 450 CfSRS – Simultaneous Ringing Service Input and Output Parameters (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
Enabled	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False
DNList	A list of up to six entries, each with the attributes below, to be simultaneously rung.			
(DNList Entry) ServiceId	A DN to be simultaneously rung.	Please refer to the definition of Forwarding Destination in Section 5.4, “Primitive Data Definitions”, on page 66.		“”
(DNList Entry) Extension	Indicates that the DN is an extension, or intercom dialing code (True). If False, the number is not an extension, but a fully qualified national numbering plan formatted number.	Boolean	True, False	False

Table 450 CfSRS – Simultaneous Ringing Service Input and Output Parameters (Sheet 2 of 2)

10.44 Station Restrictions (Business Group) (CfSR)

The Station Restrictions’ feature lets the system administrator restrict the calls permitted to and from a given station. *Originating line restrictions* refers to restrictions on calls placed from a station; *terminating line restrictions* refers to restrictions on calls being terminated to a station.

The system administrator can assign this feature to the entire business group or to individual users.

Table 451 lists the input and output parameters for this service.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
OriginatingLineRestriction Type	Indicates the restriction level for calls the subscriber originates. Must be one of these values.	enum	SemiRestrictedLine FullyRestrictedLine FullyRestrictedWithAttendant NoRestriction	NoRestriction
Terminating LineRestrictionType	Indicates the restriction level for calls terminating to the subscriber. The restriction type must be one of these values.	enum	SemiRestrictedLine FullyRestrictedLine FullyRestrictedWithAttendant NoRestriction	NoRestriction

Table 451 CfSR – Station Restriction (BG) Input and Output Parameters

10.45 Station Speed Calling (CfSPCALL)

The OpenScape Voice system-based Station Speed Calling feature provides the capability to place frequently dialed numbers in a centralized speed calling list. This feature can be provisioned at the business group level with a denied option at the subscriber level.

Attention: This feature is separate from, and not to be confused with, the System Speed Dial feature (see Section 10.46, “System Speed Dial (CfBgSpDial)”) that is only available to BGLs (not residential subscribers).

The following are the types of station speed calling:

- **One-digit station speed calling**
- **Two-digit station speed calling**

Each of the speed calling lists can be provided to a subscriber as follows:

- A private list is used by one subscriber, who can modify any entry.
- A shared list is owned by one subscriber, but can be used by many subscribers. Only the owner can modify the list entries.

A subscriber can have both a one-digit and a two-digit list. They can both be private, both be shared, or one can be private with the other one shared.

Table 452 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
CascadeListToBGLs	Indicates whether any initial list created at the FP will be copied to the subscribers to initialize the subscriber's speed dial list. The subscribers can edit their list, but it can be pre-populated with the lists from the BG. If not specified, defaults to None. The BG must have a list if it is to be cascaded to the BGLs.	enum	None One Two Both	None
ShareListWithBGLs	Indicates that the subscriber does not have his or her own list, but will use the list from the FP. The subscriber may not edit the list and any changes to the list will affect them. May be: None, One, Two. The FP must have a list if it is to be cascaded to the BGLs.	enum	None One Two	None

Table 452 CfBgSPCALL – Speed Call (Speed Dial) (BG) Input and Output Parameters (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
OneDigit	Indicates whether the subscriber has his or her own list or shares the list from another entity. This field may also contain the literal BG to indicate that the list is shared from the Business Group. Used only when this feature can be provisioned at the BGL.			
TwoDigit	Indicates whether the subscriber has his or her own list or shares a list from another entity. This field can also contain the literal BG to indicate that the list is shared from the BG. Used only when this feature can be provisioned at the BGL.			

Table 452 CfBgSPCALL – Speed Call (Speed Dial) (BG) Input and Output Parameters (Sheet 2 of 2)

10.46 System Speed Dial (CfBgSpDial)

This feature allows a station user to access pre-defined list(s) of numbers/destinations that are configured and stored at the Business Group level and have those numbers/destinations dialed by the system. The lists are accessed by the user via separately configured Prefix Access Codes, one for ListIdOne and one for ListIdTwo. The feature applies only to Business Group Lines, not to residential subscribers.

Attention: This feature is separate from, and not to be confused with, the Station Speed Calling feature (see Section 10.45, “Station Speed Calling (CfSPCALL)”) that is available at the business group level with a denied option at the subscriber level.

Table 452 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.			required
Enabled	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.			False
ListIdOne	The ID of one of the 10 possible speed dial lists configured at the BG level using the BgSpeedDialList requests. See Section 11.18, "Business Group System Speed Dial Lists" for further details on these requests. A value of 0 means that none of the lists is accessible using ListIdOne.	integer	0-10	0
ListIdTwo	The ID of one of the 10 possible speed dial lists configured at the BG level using the BgSpeedDialList requests. See Section 11.18, "Business Group System Speed Dial Lists" for further details on these requests. A value of 0 means that none of the list accessible using ListIdTwo.	integer	0-10	0

Table 453 CfBgSpDial – BG System Speed Dial Input and Output Parameters

10.47 Time-of-Day Call Forwarding (CfECF) [1]

The OpenScape Voice system-based Time-of-Day Call Forwarding feature provides a scheduling capability to use in conjunction with the call forwarding-all calls, CFBL, and CFDA features. It can also be used with the selective call forwarding feature in some circumstances. The subscriber defines the schedules, activates and deactivates the feature, and specifies the forwarding destination. Also known as Enhanced Call Forwarding.

Table 454 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.			required

Table 454 CfECF – Enhanced Call Forward Input and Output Parameters (Sheet 1 of 2)

1. Can also be referred to as Time of Day Station Call Forwarding.

Field Name	Description	Length	Range/Values	Default Value
Enabled	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False
ScreenList	An optional list of DNs to forward. A maximum of 32 entries is allowed. Screening List entries can be up to 15 digits long.	32 entries; up to 15 digits each	List of numeric strings	N/A
NumberOfRings	This parameter is obsolete and has been replaced with RingDuration, which Indicates the number of ring cycles after which (if not answered) the call will be redirected to the Voice Mail server. Assumes that the device has six-second ring cycles.	integer	1-10	4
RingDuration	Indicates the duration that the phone will ring before (if not answered) the call will be redirected to the Voice Mail server.	integer	0-60	24 seconds
ECF List	A list of ECF entries. Each entry represents a time period for a specific day of the week.			
(ECFList Entry) DayOfWeek	The day of the week to which the forwarding data defined in this entry applies.	enum	Sunday, Monday, Tuesday, Wednesday, Thursday, Friday, Saturday	Sunday
StartTime	The time at which the forwarding data defined in this entry applies, in subscriber’s local timezone.	xsd__date Time	A valid time/date string.	required
StopTime	The time at which the forwarding data defined in this entry no longer applies, in subscriber’s local timezone.	xsd__date Time	A valid time/date string.	required
CallFwdType	The type of call forwarding performed for this entry.	enum	Always, Busy, NoAnswer, BusyAndNoAnswer	Always
ScreeningList Option	Indicates which calls will be forwarded.	enum	AllNumbers ScrListNumbers NonScrListNumbers	AllNumbers
ServiceId	Indicates the destination to which calls are forwarded for the time specified in this entry.	See definition of Forwarding Destination in Section 5.4, “Primitive Data Definitions”, on page 66.		required

Table 454 CfECF – Enhanced Call Forward Input and Output Parameters (Sheet 2 of 2)

10.48 Toll and Call Restrictions (CfTRS)

The Toll and Call Restrictions feature provides destination limitations on calls originated at designated stations and private facilities.

The limitations shall be able to be provisioned through Classes of Restrictions and Alternate Classes of Restrictions, both of which shall be associated with a List of Traffic Types.

The Alternate Restriction will consist of a Class of Restriction, a Duration Switch Over Time, a Day Schedule and a Date Schedule. The Duration Switch Over specifies the activation/Deactivation of the Switch over scenario. The range of the switchover duration is between 0 to 60,000 minutes. By default, the switchover duration is 120 minutes. It will be possible to provision a predefined time every day during which the users/endpoint Profile will change from the standard Class Of Restriction to the alternate and vice versa. This provisioning will be optional.

The provisioning of the Standard Class of Restriction shall require an existing Class of Restriction Name. The provisioning of the Alternate Class of Restriction requires a Date and Day Schedule. The Date Schedule shall consist of a Starting Date with a format of:

(YYYY - MM - DD T HH : MM:SS)

and an End Time. Assistant shall be responsible for providing the administrator with the option to enter the End Date and then split the range between the Start Date and the End Date into separate Date Schedules. It is important to note that range between the Start Date and End Date cannot be greater than 14 (14 separate Date Schedules). If one Date Schedule has a range of 5 days (Assistant shall send 5 consequent Date Schedules), then it shall not be possible to create another one within the same month.

Similarly, the Day Schedule shall consist of a specific Staring Day of the Week, a Start Time and an End Time. Assistant shall be responsible for providing the administrator with the option to enter the End Day and then split the range between the Start Day and End Day into separate Day Schedules.

Validations to be performed:

1. It shall not be possible to assign Date and Day Schedules unless there is an Alternate Class of Restriction already provisioned.
2. It is possible to provision an alternate Class of Restriction without Date/Day Schedules.
3. Soap Server shall perform checks for the overlapping in Date/Day Schedules. It shall not be possible to create more than 14 Date Schedules per month.

Table 455 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
CORName	Class Of Restrictions Name	32	(1…32)	N/A
RestrictedLineList	The subscriber cannot make calls to the numbers in this list. The list may contain partial numbers, such as 212, that prevent the caller from calling any number that starts with 212. Individual entries in this list cannot be edited. The list must be completely specified.	A maximum of 10 numbers can be specified	Each entry may be a numeric string of up to 30 digits long	N/A
Alternate COR	Alternate Class of Restrictions: Consists of : AlternateCORName SwitchOverDuration Timer AlternateCORDay ScheduleList AlternateCORDate ScheduleList			
Alternate CORName	Alternate Class of restrictions	32	(1…32)	N/A
Switch Over Duration Time	Duration of the Switch over timer for activation/deactivation		0 to 60,000 minutes	0
EnableDurationTimer	Enables or disables the switchover duration to override the day schedule	Boolean	False/true	False
AllowCOSSFeatureAccessCodes	Enables users to dial COSS Feature Access Codes.	Boolean	False/true	False
ScheduleOverridesDuration	Allows users to override the schedule by dialing the feature access codes.	Boolean	False/true	false
Alternate COR DayScheduleList	Consists of up to 49 Day Schedule Entries. Each Day Schedule Entry has a Day Of Week a Start Time and Stop Time	(0 …49)		
Alternate COR DateScheduleList	Consists of up to 168 Date Schedule Entries. Each Date Schedule has a Start Date_Time and an End Time	(0…168)		

Table 455 CfTRS – Toll Restrictions Input and Output Parameters

10.49 Traffic Measurement (CfBgTM)

This feature may be provisioned for a Business Group and is applicable to the entire business group. It may not be provisioned against individual subscribers or Business Group Lines and the feature-related data is not propagated to the subscriber/BGL level.

The Traffic Measurement feature provides the ability to display traffic measurements data on a per Business Group basis. This feature can be provisioned against the BG and will not cascade to the BGLs. Unassigning this feature disables the measurement data collection.

Table 456 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range/Values	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required

Table 456 CfBgTM – Traffic Measurement Input and Output Parameters

10.50 BG Silent Monitoring (CfBgSILM)

This business group feature is responsible for counting the MonitorTagLimit and the CurrentTagUsed. The CurrentTagUsed is the current number of tags that have been used. This is a read-only field and there is no need to provision this field.

Table 457 lists the output parameters associated with this feature.

Field Name	Description	Length	Range	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. Required.			
MonitorTagLimit	Indicates the limit for monitoring subscribers per BG. If the MonitorTagLimit is 10 then only 10 subscribers can be tagged for monitoring.	Integer	Valid values 0-100	0
MaxNumOfSupervisors	Define the maximum number of allowed of Continuous monitoring supervisors and the per-Call monitoring supervisors (added via CSTA) independently. **V7.00.09.01**	Integer	0-10	5

Table 457 cfBgSLIM – BG Silent Monitoring Input and Output Parameters

10.51 Silent Monitoring Protection Feature (CfSILMProtection)

The SILMProtection is used to protect a subscriber against monitoring. Subscribers provisioned with the SILM service should be also provisioned with the SILMProtection service and this service must be enabled.

Table 458 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range	Default Value
	Supported up to V7.00.08.01 From V7.00.09.01 and upwards is replaced by SILMS/SILMA			

Table 458 cfSLIMProtection – Silent Monitoring Protection Input and Output Parameters

10.52 Silent Monitoring Feature (CfSILM)

Supported up to V7.00.08.01. From V7.00.09.01 and upwards is replaced by SILMS/SILMA.

The silent monitoring feature is provisioned for users who can monitor other users within the business group. Before a subscriber can be assigned the CfSILM feature, that subscriber's business group must be assigned the BGSILM feature. The SILMProtection and CSTA must also be assigned to the business group.

Table 459 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range	Default Value
Subscribed	Please refer to Section 5.4, "Primitive Data Definitions", on page 66. Required.			
Enabled	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.			False

Table 459 cfSLIMProtection – Silent Monitoring Input and Output Parameters

10.53 Silent Monitoring Tag Feature (CfSILMTag)

Supported up to V7.00.08.01. From V7.00.09.01 and upwards is replaced by SILMS/SILMA.

The SILMTag feature can be assigned to a BGL subscriber. Assigning this feature to a BGL subscriber means that the BGL is tagged for monitoring.

- If a Subscriber is being provisioned with SILMTag (AddReplace Operation and the value of ***Srx/Service/SilmToneIntervals = -1,*** the assignment of the Feature shall succeed but the value of the ***Periodic Tones*** shall be **false**.

- Supposing a Subscriber or a Feature Profile has the SILM Tag Feature assigned and ***EnablePeriodic Tones = true.*** If the Rtp Parameter ***Srx/Service/SilmToneIntervals changes to -1,*** Subscriber or Feature Profile shall remain as it was. (i.e. EnablePeriodicTones = true)
- If attempting to modify SILMTag on the Subscriber or Feature Profile, the ***EnablePeriodicTones*** flag shall change to ***false***.

Table 460 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. Required.			
Enabled	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False
Enable Periodic Tones	Enable/Disable periodic tone intervals	Boolean	True, False	False

Table 460 cfSLIMProtection – Silent Monitoring Protection Input and Output Parameters

10.54 Call Forwarding Restrictions (CfCFR)

When assigned to a BGL, the call forwarding restrictions (CFR) feature restricts the type of destination for which this BGL can forward calls. Its arguments represent the types of calls that can be restricted as call forwarding destinations.

Table 461 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
Enabled	Indicates whether the feature is active or inactive when billing is UsageSensitive. If not specified initially, defaults to inactive. If CfCNAM has been or is being provisioned (with Billing=UsageSensitive), the Enabled flag settings must be the same for both features.	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		False
CORName	Class Of Restrictions Name	32	(1…32)	N/A
EmergencyTraffic Type	Indicates whether call forwarding to a destination that will create an emergency call is blocked.		True, False	False

Table 461 cfCFR – Call Forwarding Restrictions Input and Output Parameters

10.55 Call Park to Server (CfCPS)

The CPS feature is applicable at the subscriber and feature profile level. Subscribers to this feature must be a business group line. The business group call transfer feature must be enabled before assigning the CPS feature to a business group line or to a feature profile with the Parking Allowed field set to true. If the call transfer feature is not active, it will not be possible to provision the CPS service. It is not possible to set both the Parking Allowed field and the Retrieval Allowed field to false.

Table 462 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. Required.			
Parking Lot Name	The name of the parking lot in which CPS service will be applied.	1 ... 20	string	required
Parking Allowed	When set to true, the administrator is able to park the call to server.	Boolean	True, False	True
Retrieval Allowed	When set to true, the administrator is able to retrieve a previously parked call.	Boolean	True, False	True

Table 462 cfCPS – Call Park to Server Input and Output Parameters

10.56 Automatic Collect Call Blocking for Brazil

The new ACCB service will work (only) with AudioCodes Gateways or HiPath4000/RG8300 that already include the necessary ACCB signaling procedures to work within the Brazilian PSTN. The ACCB feature is only applicable to the original called OSCV subscriber. If the call is forwarded or redirected (transferred) any ACCB service assigned to the new target subscriber is not applied i.e. if the original called OSCV subscriber does not have ACCB but the new target (due to forwarding/transfer) does have ACCB then ACCB will not be applied.

OpenScape Voice administrators may wish to automatically block incoming collect calls to some OpenScape Voice users but still allow other users to receive collect calls and make their own choice to accept or reject the call.

Table 463 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. Required.			

Table 463 ACCB – Automatic Collect Call Blocking for Brazil Input and Output Parameters (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
Enabled	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			False

Table 463 ACCB – Automatic Collect Call Blocking for Brazil Input and Output Parameters (Sheet 2 of 2)

10.57 Openscape Mobile Service (OSMO)

OpenScape Server has the capability to provision OpenScape Mobile (OSMO) clients. Such type of clients allows mobile workers to register with Openscape Voice with their own Address of Record and manage calls through the mobile client.

The basic configuration for an Openscape Mobile user is:

1. A desk phone. The desk phone is configured with the user's ONS number. The desk phone may be located on campus with direct connectivity to OpenScape Voice or in a home office with access to OSV through an SBC.
2. A dual band Android mobile phone capable of running the Unify Openscape Mobile application.
3. The Openscape Mobile VoIP agent will be provisioned with its own DN (Openscape Mobile WiFi OND), and static ONS binding in OSV.
4. Cellular phone number.
5. An Xpressions voicemail box.

An Openscape Mobile user shall only expose his ONS number for all his communications. This includes calls made or received from the user's DeskPhone, Mobile over WiFi, or Mobile over Cellular. Step 1 provides inbound ONS service, and outbound ONS service over WiFi.

The mobile device is the user's remote control for his enterprise communications. While connected over WiFi with Openscape Mobile the user can control device selection for incoming calls:

1. Answer call on Mobile over WiFi
2. Answer call on Mobile over Cellular
3. Answer call on DeskPhone
4. Decline Call
5. Send to voicemail

The basic routing for incoming is to successively send the call to:

1. The mobile device over WIFI
2. The desk phone
3. The mobile device over Cellular
4. The Xpression voice mail

While connected over WiFi with Openscape Mobile, the user can handover an active ONS call (call in connected state) between his mobile WiFi, mobile Cellular, and desk phone devices.

Android Use cases:

1. Local Campus/Corporate WiFi.

   Mobile client and Deskphone register with OSV using SIP.

2. WiFi hotspot.

   Mobile client registers with OSV using SIP through SBC.

3. Remote Campus/ Corporate WiFi.

   Mobile client registers with OSV using SIP.

4. Home Office/ Home WiFi.

   Mobile client and Deskphone register with OSV using SIP through SBC.

I**mportant:**

To successfully assign OSMO service, the subscriber must NOT:

- "Belong to a feature profile
- "Be part of an MLHG
- "Have other services assigned
- "Be an Executive/Assistant Group member
- "Has keyset provisioned
- "Its Deskphone is already associated with another OSMO subscriber
- "Be profile only subscriber
- "Be a call pick up group member
- "Be a community group member
- "Define as a Main Device which is already associated with an existing OSMO.
- Define as Main Device, a subscriber DN which has OSMO service assigned

Additionally, to successfully assign OSMO service, the subscriber MUST:

- Have ExternalDn flag set to false
- Have SIP security enabled
- Send Insecure referred by Header flag set to true

For provisioning of OSMO please refer to section 9.1 "Create Subscriber" and 9.23 "Update Mobile Info"

Table 464 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
V5.00.20.01		subRC_APP_FEAT_OSMO_ONS_NOT_ASSIGNED_TO_DESK_PHONE_DN subRC_APP_FEAT_OSMO_DESK_PHONE_NUMBER_IN_USE subRC_APP_NOT_ALLOWED_REMOVING_ONS_WITH_MOBILEDN_REFERENCE subRC_APP_SUB_OSMO_FEATURES_CONFLICT subRC_APP_FEAT_OSMO_CSTA_NOT_ASSIGNED_TO_DESK_PHONE_DN subRC_APP_FEATURES_AND_OSMO_CONFLICT
V7		subRC_APP_OSMO_EXTERNAL_DN_FLAG_CONFLICT subRC_APP_EP_INSECURE_REFERRED_BY_HEADER_OSMO_CONFLICT subRC_APP_SUB_ROUTETOCELL_CELLDEVICENUMBER_CONFLICT

Table 464 OSMO Service Provisioning Return Codes

10.58 Routing Service

The purpose of this service is to expose CSTA Routing registration and call control services and meet the basic functional requirements defined in the ECMA 269 CSTA standard.

Table 465 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. .			required

Table 465 Routing Service Input and Output Parameters (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
Enabled	Indicates whether the feature is active or inactive when billing is UsageSensitive. If not specified initially, defaults to inactive. If CfCNAM has been or is being provisioned (with Billing=UsageSensitive), the Enabled flag settings must be the same for both features	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.		False

Table 465 Routing Service Input and Output Parameters (Sheet 2 of 2)

10.59 Network Wide Call Park to Server (Endpoint Profiles)

Network Wide Call Park to Orbit is an extension of the existing feature Call Park to Server introduced with FRN3293. While Call Park to Server is applicable to subscriber level and Feature profile level only, NW Call Park to Orbit is applicable to Endpoint Profiles.

NW Call Park to Orbit shares the same logic and structure of Call Park to Server:

Thus the following data will be provisioned with CPO feature:

- Parking allowed: YES/NO (default is YES)
- Retrieval allowed: YES/NO (default is YES)
- Parking Lot: The name of the Parking Lot in which CPO service will be applied.
- Business group: The business group which contains the Parking Lot.

Administrator provisions the feature by defining an existing parking lot and setting Boolean flags Park and Retrieve allow. The scope of this feature is to provide users outside the Openscape Voice with the capability of parking and retrieving calls to/from a central parking lot.

Prerequisites/Restrictions:

- Setting both fields **Parking allowed** and **Retrieval allowed** to false is not allowed.
- Network Wide Call Park to Server is only applicable to endpoint profiles that are assigned to private numbering plans. Any attempt to assign it to E164NANP endpoint profile will issue an error.

- Network Wide Call Park to Server is only applicable to an endpoint profile that is inside a business group.

Validation for inappropriate combinations of Business groups and Parking Lots will be performed before assigning the feature.

Table 466 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. .			required
Parking Lot Name	The name of the parking lot in which CPS service will be applied	1-20	String	Required
Parking Allowed	When set to true, the administrator is able to park the call to server.	Boolean	True,False	True
Retrieval Allowed	When set to true, the administrator is able to retrieve a previously parked call.	boolean	True,False	True

Table 466 NWfCPS – Network Wide Call Park to Server Input and Output Parameters

10.60 CSTA Application external device name provider (Endpoint Profiles)

CSTA Network Interface Monitoring service (CNIM). CNIM enables the support CSTA on OSV Network Interface Devices (NID). CNIM supports a subset of services and events that can be used by CSTA-enabled adjunct application to detect the presence of an inbound and outbound off-network call; lookup a name string based on the calling/called device number and associate the calling/called device name within the OSV call context via CSTA service. CNIM is applicable to Endpoint Profiles.

The Endpoint page in CLI and Assistant shall be enhanced to present the CSTA NID deviceID. The NID shall be presented in the Device Number (DeN) format. DeN format is a non-dialable 4-byte integer value. OpenScape Voice shall adopt the DeN format used by HiPath4000. HiPath4000 uses the most significant byte to identify the device type and the least significant 3-bytes for the device number. The device number shall be based on the least significant 3-bytes of the existing internal endpoint identifier (EPID) assigned in the endpoint database record. The internal EPID is updated by one each time an endpoint device is added to the system. EPID values are not re-used when an endpoint is deleted. Please refer to Endpoints sections for details regarding the new data returned.

Table 467 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range	Default Value
NameProvider	**V7.00.01.10** This option is used to activate / deactivate CSTA services that enable applications to associate a calling device name with the off-net calling device.	boolean	True / False	True

Table 467 CNIM – CSTA Application external device name provider Input and Output Parameters

10.61 Silent Monitoring Supervisor Feature (CfSILMS)

The SILMS feature can be assigned to a BGL subscriber. Assigning this feature allows users to monitor other users. The monitoring is continuous, regardless they are in a call or not. The monitoring session remains active as long as user desires and not for a specific call. Users provisioned with SILMS, gain the capability of monitoring other users that have SILM Agent feature provisioned.

- HoldCallWhenInactive, InactiveIntercept and SecureConnection are provisionable only when SilentMonitoring = True

Table 468 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 65. Required.			
Enabled	Please refer to Section 5.4, “Primitive Data Definitions”, on page 65.			False
SilentMonitoring	Indication of whether a Supervisor is allowed to invoke Silent Monitoring	Boolean	True, False	True
Barge-In	Indication of whether a Supervisor is allowed to invoke Barge-In	Boolean	True, False	True
Whisper	Indication of whether Supervisor is allowed to invoke whisper from a silent monitoring session. Applicable when SilentMonitoring=True V7.00.23.01 and forward.	Boolean	False, True	False

Table 468 SILMS – Silent Monitoring Supervisor Input and Output Parameters

Field Name	Description	Length	Range	Default Value
Monitoring Sessions Data	Available when **SilentMonitoring** = True			
Local		CfSILMS-locPres	• Enabled • SilentMonitoring • BargeIn • HoldCallWhenInactive • SecureConnection • InactiveIntercept	
HoldCallWhenInactive	Indication of whether Supervisor shall be placed on hold when the monitoring session is inactive	Boolean	True, False	False
InactiveIntercept	The provisioned intercept for the inactive monitoring session	String	list of character strings , length : 1-15 or empty string	MSSLM_SESSION_INACTIVE
SecureConnection	Indication of whether Supervisor connection with Media Server shall be considered secure	Boolean	True, False	False

Table 468 SILMS – Silent Monitoring Supervisor Input and Output Parameters

10.62 Silent Monitoring Agent Feature (CfSILMA)

The SILMA feature can be assigned to a BGL subscriber. Assigning this feature allows users to be monitored by other users. The monitoring is continuous, regardless they are in a call or not. The monitoring session remains active as long as user desires and not for a specific call. Users provisioned with SILMA, can be monitored by other users that have SILM Supervisor feature provisioned.

- SILMTag and ToneMonitorActivate provisionable only when SilentMonitoring = True
- ToneMonitorInterval and NotifyPartnerDevices provisionable only when ToneMonitorActivate = true

Table 469 lists the input and output parameters associated with this feature.

Field Name	Description	Length	Range	Default Value
Subscribed	Please refer to Section 5.4, “Primitive Data Definitions”, on page 65. Required.			

Table 469 SILMS – Silent Monitoring Agent Input and Output Parameters

Field Name	Description	Length	Range	Default Value
Enabled	Please refer to Section 5.4, “Primitive Data Definitions”, on page 65.			False
SilentMonitoring	Indication of whether a Supervisor is allowed to invoke Silent Monitoring	Boolean	True, False	True
Barge-In	Indication of whether a Supervisor is allowed to invoke Barge-In	Boolean	True, False	True
WhisperTo	Indication of whether Agent is allowed to be whispered-to Applicable when SilentMonitoring=True V7.00.23.01 and forward.	Boolean	False, True	False
SILMTag	Indication of whether Tone Monitoring is enabled. Applicable when SilentMonitoring=True	Boolean	True, False	False
Tone Monitoring Data				
ToneMonitorActivate	Indication of whether Tone Monitoring is enabled. Applicable when SilentMonitoring=True	Boolean	True, False	False
ToneMonitorInterval	Indication of interval in seconds in which the tone shall be played to the Agent or the partner devices. A value of 0 specifies utilization of	Integer	[0…100]	30
NotifyPartnerDevices	Notify Partner Devices. Applicable when ToneMonitorActivate =True	Boolean	True, False	True

Table 469 SILMS – Silent Monitoring Agent Input and Output Parameters

11 Business Groups

A Business Group (BG) is an entity of related subscribers. (Subscribers may optionally belong to a business group.) It is necessary to provide a BG Name in the CreateSubscriber request in order to assign the subscriber to the BG. When subscribers are created with an associated BG, they are also referred to as Business Group Lines (BGLs). Please refer to Section 4.9, “Business Groups”, on page 52 for more information.

11.1 Create BG

Used to create a Business Group. When creating and deleting a Business Group, the system generated-CustomerId from the Customer table will be re-used for the Business Group Index. The Business Group Name is stored as the Customer Name.

The maximum number of Business Groups supported on the system is controlled by an RTP parameter, Srx/Main/MaxNumberBGSupported, with a predefined upper limit of 6000.

Table 470 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.02.01 11.00.01		supersuccessful subRC_APP_BG_NAME_LEN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_DISPLAY_NUM_LEN_INVALID subRC_APP_PNP_NAME_LEN_INVALID subRC_APP_BG_NUMBER_PLAN_DOES_NOT_EXIST subRC_DB_BG_NUMBER_PLAN_LOOKUP_FAILED subRC_APP_BG_NUMBER_PLAN_OUT_OF_RANGE subRC_APP_BG_NUMBER_PLAN_DUPLICATE subRC_DB_BG_NUMBER_PLAN_LOOKUP_FAILED subRC_APP_BG_OUT_OF_RANGE subRC_APP_BG_DUPLICATE subRC_DB_BG_INSERTION_FAILED subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_SERVICES_MODIFY_FAILED

Table 470 Create BG Return Codes

Table 471 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
BGName	Business Group name.	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		

Table 471 Create BG Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
NumberPlan Name	Private Number Plan. Refer to Section 5.4, “Primitive Data Definitions”, on page 66.	1-31	Character String	System Default NP PSTN – E164NANP – PNP DEFAULT
Display Number	When a subscriber with only a private number makes a external BG call, the BG’s DisplayNumber is presented as the calling number.	15	Numeric string	required
BGAccessCode	Indicates the BG Access Code for the specifig BG.	1..15	Numeric string (0..9), plus * and #.	N/A (Not mandatory field)

Table 471 Create BG Input Fields (Sheet 2 of 2)

11.2 Update BG Status

This method updates a BG’s status.

Table 472 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
01.01.01		subRC_SUCCESS subRC_APP_BG_NAME_LEN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_DB_BG_MODIFICATION_FAILED

Table 472 Update BG Status Return Codes

Table 473 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
BGName	Business Group Name.	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
BGStatus	The status of the Business Group.	enum	Active, Blocked	N/A

Table 473 Update BG Status Input Fields

11.3 Add BG Main Number

The AddBGMainNumber method adds the Main Number for the BG, with Attendant Number, auto Attendant Number and Auto Attendant Status. The Main Number can be a virtual number or real number. The Attendant must be created before it can be assigned to the BG.

Table 474 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
01.01.01		subRC_SUCCESS subRC_APP_BG_NAME_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_MAIN_NUM_LEN_INVALID subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_MAIN_NUM_DOES_NOT_EXIST subRC_DB_BG_MAIN_NUM_LOOKUP_FAILED subRC_APP_BG_MAIN_NUM_ATTENDANT_LEN_INVALID subRC_DB_BG_MAIN_NUM_ATTENDANT_LOOKUP_FAILED subRC_APP_BG_MAIN_NUM_AUTOATT_LEN_INVALID subRC_APP_BG_MAIN_NUM_AUTOATT_MISSING subRC_APP_BG_MAIN_NUM_AUTOATT_MISSING subRC_DB_BG_MAIN_NUM_MODIFY_FAILED

Table 474 Add BG MainNumber Return Codes

Table 475 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
BGName	Business Group Name.	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
BGMainNumber	The Main Number to be added. For further details about Main Numbers, please refer to Section 4.10.1, “Main Numbers”, on page 52.	30	Numeric string	required
BGMainNumberAttendant	The BG Attendant to which calls to the Main Number are directed. For further details about BG Attendants, please refer to Section 4.10.2, “Attendant Numbers”, on page 53.	30	Numeric string	“”

Table 475 Add BG Main Number Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
AutoAttendantNumber	The number for redirecting calls that terminate at the main number if the auto-attendant service is available. If the auto-attendant flag is set to true, then the auto-attendant number must be specified. For further details about the auto-attendant service, please refer to Section 4.10.1, "Main Numbers", on page 52.	30	Numeric string	""
AutoAttendantStatus	Indicates auto-attendant availability.	enum	Enabled, Disabled	Disabled

Table 475 Add BG Main Number Input Fields (Sheet 2 of 2)

11.4 Delete BG Main Number

The DeleteBGMainNumber method is used to remove a Main Number from a BG.

Table 476 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
01.01.01		subRC_SUCCESS subRC_APP_BG_NAME_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_MAIN_NUM_LEN_INVALID subRC_APP_BG_MAIN_NUM_DOES_NOT_EXIST subRC_DB_BG_MAIN_NUM_LOOKUP_FAILED subRC_DB_BG_MAIN_NUM_ASSIGN_FAILED subRC_INTERNAL_ERROR

Table 476 Delete BG Main Number Return Codes

Table 477 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
BGName	Business Group Name.	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.		
OperatorId	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.			
BGMainNumber	The Main Number to be added. For further details about Main Numbers, please refer to Section 4.10.1, "Main Numbers", on page 52.	30	Numeric string	required

Table 477 Delete BG Main Number Input Fields

11.5 Update BG Main Number

The UpdateBGMainNumber method is used to add, replace, or delete a BG's main number.

Table 478 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
01.01.01		subRC_SUCCESS subRC_APP_BG_NAME_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_MAIN_NUM_LEN_INVALID subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_MAIN_NUM_DOES_NOT_EXIST subRC_DB_BG_MAIN_NUM_LOOKUP_FAILED subRC_APP_BG_MAIN_NUM_ATTENDANT_LEN_INVALID subRC_DB_BG_MAIN_NUM_ATTENDANT_LOOKUP_FAILED subRC_APP_BG_MAIN_NUM_AUTOATT_LEN_INVALID subRC_APP_BG_MAIN_NUM_AUTOATT_MISSING subRC_APP_BG_MAIN_NUM_AUTOATT_MISSING subRC_DB_BG_MAIN_NUM_MODIFY_FAILED

Table 478 Update BG Main Number Return Codes

Table 479 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
BGName	Business Group Name.	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.		
OperatorId	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.			
BGMainNumber	The Main Number to be added. For further details about Main Numbers, please refer to Section 4.10.1, "Main Numbers", on page 52.	30	Numeric string	required
BGMainNumberAttendant	The BG Attendant to which calls to the Main Number are directed. For further details about BG Attendants, please refer to Section 4.10.2, "Attendant Numbers", on page 53.	30	Numeric string	N/A

Table 479 Update BG Main Number Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
AutoAttendantNumber	The number for redirecting calls that terminate at the main number if the auto-attendant service is available. If the auto-attendant flag is set to true, then the auto-attendant number must be specified. For further details about the auto-attendant service, please refer to Section 4.10.1, "Main Numbers", on page 52.	30	Numeric string	N/A
AutoAttendantStatus	Indicates auto-attendant availability.	enum	Enabled, Disabled	N/A

Table 479 Update BG Main Number Input Fields (Sheet 2 of 2)

11.6 Update BG Features

The UpdateBGFeatures method is used to add to, replace, or delete a feature from a Business Group. Only a small number of features apply at the BG level, and these are features that only apply at that level - that is, the associated data is not propagated to the BGLs and is not displayed at the BGL level, although the data associated with the BG features is used during call processing for the BGLs where applicable.

Table 480 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
01.01.01		subRC_SUCCESS subRC_APP_FEAT_FEATURE_NOT_ENABLED subRC_APP_BG_FEATURE_DENIED subRC_APP_BG_NAME_LEN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_DB_BG_CUSTOMER_TABLE_LOOKUP_FAILED subRC_APP_BG_FEAT_BGE911LIN_DEFAULT_INT_MISSING subRC_APP_BG_FEAT_BGE911LIN_EMERGENCY_LEN_INVALID subRC_APP_BG_FEAT_BGE911LIN_EMERGENCY_DOES_NOT_EXIST subRC_APP_BG_FEAT_BGE911LIN_EMERGENCY_LOOKUP_FAILED subRC_APP_BG_FEAT_BGKS_MAN_HOLD_REC_TIME_OUT_OF_RANGE subRC_APP_BG_FEAT_BGKS_LINE_RES_TIME_OUT_OF_RANGE subRC_APP_BG_FEAT_NBCPU_ID_OUT_OF_RANGE subRC_APP_BG_FEAT_NBCPU_ID_MISSING subRC_APP_BG_FEAT_HD_AUTO_LOGOFF_TIME_INVALID subRC_APP_BG_DELETE_BGLS_HD_ASSIGNED subRC_SYS_BG_LC_MAX_CONF_MEMBERS subRC_APP_BG_LC_MIN_PORTS_OUT_OF_RANGE subRC_APP_BG_LC_MIN_PORTS_NOT_AVAILABLE subRC_APP_BG_LC_MIN_MAX_PORTS_CONFLICT subRC_APP_BG_LC_MAX_PORTS_OUT_OF_RANGE subRC_APP_BG_LC_MAX_PORTS_NOT_AVAILABLE subRC_APP_BG_LC_MAX_CONF_OUT_OF_RANGE subRC_APP_BG_LC_MAX_CONF_MEMBERS_OUT_OF_RANGE subRC_APP_BG_LC_CONF_RECALL_DISC_TIMER_OUT_OF_RANGE subRC_APP_BG_LC_MAX_CONF_MAX_PORT_INVALID subRC_APP_BG_LC_MAX_CONF_MEMBERS_MAX_PORT_INVALID subRC_APP_BG_FEAT_BGLC_DOES_NOT_EXIST subRC_APP_BG_LC_SCC_ASSIGNED

Table 480 Update BG Features Return Codes

Table 481 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
BGName	Business Group Name.	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
ListOfBGFeatures	The list of features to be modified.		A list of pointers to the data structures of all the features to be modified. The supported features are discussed in Chapter 10, “Features”	N/A

Table 481 Update BG Features Input Fields

11.7 Get BG Info

The GetBGInfo method is used to retrieve BG-related information, such as BG Name, Main Number, and Dialing Plan, as well as the features that are assigned to the BG.

Table 482 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
01.01.01 04.02.01		subRC_SUCCESS subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_DB_BG_CPU_LOOKUP_FAILED subRC_APP_BG_CPU_ID_INVALID subRC_APP_BG_ATT_NUM_LOOKUP_FAILED

Table 482 Get BG Info Return Codes

Table 483 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
BGName	Business Group Name.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		
InterfaceWSDL	Refer to Section 5.1.1, “The Interface WSDL and Interface Release Control”, on page 61.			

Table 483 Get BG Info Input Fields

11.7.1 Get BG Info By Options

This method returns the BG info based on options being options specified in the SOAP request. The options include a list of features, CPUs, MainNumber and Number Plan, and BGService Access Codes.

Table 484 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.02.01		subRC_SUCCESS subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_DB_BG_CPU_LOOKUP_FAILED subRC_APP_BG_CPU_ID_INVALID subRC_APP_BG_ATT_NUM_LOOKUP_FAILED

Table 484 Get BG Info By Options Return Codes

Table 485 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
BGName	Business Group Name.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		
Interface Release	The release version being used by the requesting SOAP client.	Please refer to the definition of InterfaceRelease in Section 5.4, “Primitive Data Definitions”, on page 66.		Latest
InterfaceWSDL	The WSDL version being used by the requesting SOAP client.	For more information, refer to Section 5.1.1, “The Interface WSDL and Interface Release Control”, on page 61.		
GetBGData Options	Information to be included for display	Structure of Enum	IncludeCpu__, IncludeListOfFeatures__ IncludeBgMainNumberList__ IncludeBgTrafficInfo__	ALL

Table 485 Get BG Info By Options Input Fields

11.8 Get BG List

The GetBGList method is used to retrieve the list of BGs created on the system. Filtering is performed on NumberPlanName, BGName (supporting wildcarding) and Resource Management Type. If Number Plan Name is provided, only BGs that are associated with the given Numbering Plan are returned.

Sorting is also supported by BGName if the GetBGListSortList list is filled. ascending/descending ordering option is provided. If left empty sorting is by default on bg_index.

Table 486 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
01.01.01 04.02.01		subRC_SUCCESS subRC_DB_BG_LIST_COUNT_LOOKUP_FAILED subRC_SYS_BG_LIST_ALLOC_ERR subRC_DB_BG_LOOKUP_FAILED

Table 486 Get BG List Return Codes

Table 487 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required

Table 487 Get BG List Input Field Descriptions (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			N/A
BGName	Business Group Name. Optional. Provided for filtering. A string containing the wildcard character (%) can also be the input. Matching entries will be returned in this case.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		“”
NumberPlanName	The name of the Private Numbering Plan on which to filter. Only BGs that are associated with this Number Plan will be returned.	1..31	Character string.	N/A
DefaultResrcMgmtType	Specifies the Resource Management Type of the QoS Profile.	enum		N/A
GetBGListSortList.GetBGListSortType	Provides selection on sorting type. Currently only sorting by BGName is provided.	enum	BGListSortBGName	N/A
GetBGListSortList	Provides selection of sorting order	enum	SortOrderAscending, SortOrderDescending	SortOrderAscending

Table 487 Get BG List Input Field Descriptions (Sheet 2 of 2)

11.9 Delete BG

The DeleteBG method is used to delete a BG. The BG cannot have any existing subscribers or endpoint profiles or associated numbering plans or BG subnets. Other BG-dependent objects, such as BG departments, are deleted automatically upon successful completion of this request.

Table 488 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes	
V4.00.04.01		subRC_SUCCESS	
		subRC_APP_BG_NAME_INVALID	
		subRC_APP_COMM_OPERATOR_INVALID	
		subRC_APP_BG_DOES_NOT_EXIST	
		subRC_DB_BG_LOOKUP_FAILED	
		subRC_APP_BG_SUB_IN_USE	

Table 488 Delete BG Return Codes (Sheet 1 of 2)

WSDL Version	Obsolete in WSDL Version	Return Codes	
		subRC_APP_BG_DP_IN_USE	
		subRC_DB_BG_EP_PROFILE_LOOKUP_FAILED	
		subRC_APP_BG_EP_PROFILE_IN_USE	
		subRC_DB_BG_SUBNET_LOOKUP_FAILED	
		subRC_APP_FP_LIST_LOOKUP_FAILED	
		subRC_APP_FP_DELETION_FAILED	
		subRC_DB_BG_DEPT_NUM_LOOKUP_FAILED	
		subRC_DB_BG_DEPT_DELETION_FAILED	
		subRC_APP_BG_AUTH_CODE_DELETION_FAILED	
		subRC_APP_BG_SERVICES_MODIFY_FAILED	
		subRC_DB_BG_MAIN_NUM_DELETION_FAILED	
		subRC_DB_BG_DELETION_FAILED	
		subRC_APP_BG_PKLOT_QUERYLIST_FAILURE	30301062 Delete BG was aborted. Failure while retrieving Parking Lot List
		subRC_APP_BG_PARKING_LOT_EXIST	30301061 Delete BG was aborted.BG has Parking Lot(s) associated with it

Table 488 Delete BG Return Codes (Sheet 2 of 2)

Table 489 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
BGName	Business Group Name.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			

Table 489 Delete BG Input Fields

11.10 Get BG Attendant Numbers

The GetBGAttendantNumbers method is used to retrieve the list of DNs provisioned as Attendant Numbers for the BGs.

Table 490 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
03.02.08		subRC_SUCCESS subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_DB_BG_ATT_LOOKUP_FAILED

Table 490 Get BG Attendant Numbers Return Codes

Table 491 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
BGName	Business Group Name.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			N/A

Table 491 Get BG Attendant Numbers Input Fields

11.11 Delete BG Cpu

This method deletes the list of CPUs (call pickup groups) that belong to this BG. All the BGLs that belong to this BG and that have the corresponding CPUId will have their CPUIds set to zero (0). All the BGLs will have this subscriber feature set to inactive.

Table 492 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.01.14		subRC_SUCCESS subRC_APP_BG_CPU_ID_INVALID

Table 492 Delete BG Cpu Return Codes

Table 493 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
BGName	Business Group Name.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
CpuId	The ID of the Call Pickup Group to remove.	See definition of CpuId in Section 9.1, “Create Subscriber”, on page 303.		

Table 493 Delete BG Cpu Input Fields

11.12 Update BG Parms

This request is used to modify the values of BG-related parameters for an existing Business Group.

Table 494 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.02.07		subRC_SUCCESS subRC_APP_BG_NAME_LEN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DISPLAY_NUM_LEN_INVALID subRC_DB_BG_MODIFICATION_FAILED subRC_APP_BG_MDR_CUST_ID_LEN_INVALID subRC_APP_BG_MDR_CUST_ID_MISSING subRC_APP_BG_MDR_CUST_ID_DUPLICATE subRC_DB_BG_MDR_CUST_ID_LOOKUP_FAILED subRC_APP_BG_NUM_PLAN_MOD_NOT_ALLOWED subRC_APP_BG_NUMBER_PLAN_NAME_LEN_INVALID subRC_APP_BG_NUMBER_PLAN_DOES_NOT_EXIST subRC_APP_PNP_LOOKUP_FAILED subRC_APP_BG_NUMBER_PLAN_DUPLICATE subRC_DB_BG_NUMBER_PLAN_LOOKUP_FAILED subRC_DB_BG_CUSTOMER_TABLE_LOOKUP_FAILED subRC_APP_BG_MEDIA_SVR_RETRY_OUT_OF_RANGE subRC_APP_BG_MEDIA_SVR_RETRY_OUT_OF_RANGE subRC_APP_BG_INPUT_RETRY_OUT_OF_RANGE

Table 494 Update BG Parms Return Codes

Table 495 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
BGName	Business Group Name.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
MDRIndicator	Message Detail Recording Indicator. This parameter is set as On/Off to indicate the insertion of the MDR Customer ID in the CDR records. To enable this feature, the MDR Customer ID must be provided.	enum	Off, On	Off

Table 495 Update BG Parms Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
MDRCustID	Message Detail Recording Customer ID. A string of digits associated with a BG that must be unique in the switch.	1..10	Numeric string	N/A
NumberPlanName	The new Number Plan Name.	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		N/A
BGCommonNumberPlanName	The name of a common numbering plan to be used when BGLs “escape” from a BG private numbering plan. Only one BGCommonNumberPlanName is allowed per BG, and it is optional. Note that the specified numbering plan must be reserved for the BG before it can be assigned as the BG Common Numbering Plan.	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		N/A
DisplayNumber	The new Display Number.	See definition of DisplayNumber in Section 11.1, “Create BG”, on page 429.		N/A
MediaSvrRetry Count	Media Server Retry Count. Specifies the number of times to re-try a failed media server request for digit collection. Specific to the HotDesking feature.	integer	1..5	3 (on BG creation - for this request, N/A)
MediaSvrTimeout Interval	Media Server Timeout Interval. Timeout period for media server requests. Specific to the HotDesking feature.	integer	1..60	15 (on BG creation - for this request, N/A)
InputRetryCount	Input Retry Count. The maximum number of times to retry in response to invalid subscriber input. Specific to the HotDesking feature.	integer	1..5	3 (on BG creation - for this request, N/A)
NewBGName	The new BG Name	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		
BGAccessCode	Indicates the BG Access Code for the specifig BG.	1..15	Numeric string (0..9), plus * and #.	N/A (Not mandatory field)

Table 495 Update BG Parms Input Fields (Sheet 2 of 2)

11.13 Business Group Departments

Business Groups can further be divided into departments and BGLs can then be assigned to them. The following functionality for departments is supported.

- Create department — Every BG can have a maximum of 50 departments, with any number of BGLs assigned to them. The name of a department is restricted to twenty characters.
- Delete department — Departments which do not have any BGLs assigned to them can be deleted. If a department has BGLs assigned to them the request for deletion will be rejected.
- Modify department name — Change the name of an existing department.
- Retrieve a list of departments — This returns the list of departments created for the particular BG.
- Retrieve a list of subscribers assigned to a department.
- Assign/Unassign a BGL to a department.

11.13.1 Create BG Dept

This method is used to create a BG Department.

Table 496 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01.06		subrc_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_DEPT_NAME_LEN_INVALID subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DEPT_OUT_OF_RANGE subRC_DB_BG_DEPT_NUM_LOOKUP_FAILED subRC_APP_BG_DEPT_DUPLICATE subRC_APP_BG_DEPT_EXIST subRC_DB_BG_DEPT_INSERTION_FAILED

Table 496 Create BG Dept Return Codes

Table 497 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
BGName	The name of an existing BG for which a department is being created.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		Null

Table 497 Create BG Dept Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
BGDeptName	The name of the department being created. Limited to 50 departments for a single BG.	1..20	Character string.	Null

Table 497 Create BG Dept Input Fields (Sheet 2 of 2)

11.13.2 Delete BG Dept

This method enables you to delete a BG department.

Table 498 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01.06	N/A	subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_DEPT_NAME_LEN_INVALID subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_DB_BG_DEPT_DOES_NOT_EXIST subRC_DB_BG_DEPT_LOOKUP_FAILED subRC_APP_BG_DEPT_IN_USE subRC_DB_BG_DEPT_NUM_LOOKUP_FAILED subRC_DB_BG_DEPT_DELETION_FAILED

Table 498 Delete BG Dept Return Codes

Table 499 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
BGName	The name of an existing BG for which a department is being created.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		Null
BGDeptName	The name of the department being created. Limited to 50 departments for a single BG.	1..20	Character string.	Null

Table 499 Delete BG Dept Input Fields

11.13.3 Modify BG Dept

This method adds, modifies, or removes a BG department.

Table 500 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01.06		subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_DEPT_NAME_LEN_INVALID subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_DB_BG_DEPT_DOES_NOT_EXIST subRC_DB_BG_DEPT_LOOKUP_FAILED subRC_APP_BG_DEPT_DUPLICATE subRC_DB_BG_DEPT_MODIFY_FAILED

Table 500 Modify BG Dept Return Codes

Table 501 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
BGName	The name of an existing BG for which a department is being created.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		Null
BGDeptName	The name of the department being created. Limited to 50 departments for a single BG.	1..20	Character string.	Null

Table 501 Modify BG Dept Input Fields

11.13.4 Get BG Dept List

This method retrieves a list of BGs.

Table 502 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01.06		subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_DEPT_NAME_LEN_INVALID subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_DB_BG_DEPT_LOOKUP_FAILED subRC_SYS_BG_DEPT_ALLOC_ERR

Table 502 Get BG Dept List Return Codes

Table 503 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
BGName	The name of an existing BG for which a department is being created.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		Null
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			N/A

Table 503 Get BG Dept List Input Fields

11.14 Business Group Subnets

Business group subnet administration provides the BG administrator with the ability to provision the E9-1-1 Subnet Table on a per-BG basis. The purpose of a BG subnet is to associate one or more LINs (Location Identification Numbers) with a particular physical location to be used by an emergency operator, also known as a PSAP (Public Safety Answering Point). The physical location is expressed with either an IP address/ Subnet mask and/or Location Domain name. Location Domain and IP address/Mask must be unique. It is not allowed to create a Bg Subnet which either IP/Mask or Location Domain is already provisioned. Registered endpoints may have their own associated LIN that determines not only the BG subnet but also the caller that placed an emergency call. Initially, the caller starts an E911 call. After OpenScape Voice identifies the called digits as an emergency number, it retrieves a LIN from the list and the corresponding route number. Its Address Of Record is also stored to allow callbacks from Public Safety Answering Point (PSAP). The route number is used to lead the call to an appropriate gateway, which routes the call to the appropriate PSAP center. The PSAP center uses the LIN as a key to trigger its database (ALI) and retrieve correct information on the caller's location. If PSAP is not available, the call can be redirected to a default callback number (DCBN entry). Originally, PSAP uses the callback number (CBN entry) stored in the ALI database. OpenScape Voice identifies an incoming call from the PSAP as an emergency call and checks who has to be called by using the AoR. This call bypasses all features that prevent a call from being delivered to the station that placed an E911

emergency call. The callback number must be defined for each LIN entry, but a default callback number does not. It is not possible to provisioning a callback number that is already a registered DN, and vice versa. Only a registered DN can be used as a valid default callback number. With the SOAP requests described in the following section, the SOAP server provides the ability to create, update, delete, and retrieve information about BG subnets.

Attention: BG Subnets are intended for use in the Enterprise environment.

11.14.1 Get Bg Subnet Info

This method retrieves information about a BG's subnets.

Table 504 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
V5.00.18.01		subRC_APP_DSA_LOCATION_DOMAIN_NAME_SIP_PROXY_INVALID
05.03.01		subRC_SUCCESS subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_SUBNET_OR_IP_LEN_INVALID subRC_APP_BG_SUBNET_IP_INVALID subRC_APP_BG_SUBNET_MASK_LEN_INVALID subRC_APP_BG_SUBNET_MASK_INVALID subRC_APP_BG_SUBNET_DOES_NOT_EXIST subRC_DB_BG_SUBNET_LOOKUP_FAILED
V4.00.01.08		subRC_SUCCESS subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_SUBNET_OR_IP_LEN_INVALID subRC_APP_BG_SUBNET_IP_INVALID subRC_APP_BG_SUBNET_MASK_LEN_INVALID subRC_APP_BG_SUBNET_DOES_NOT_EXIST subRC_DB_BG_SUBNET_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_SUBNET_MEMORY_ALLOCATION_ERROR

Table 504 Get Bg Subnet Info Return Codes

Table 505 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default Value
BGName	Business Group Name.	Please refer to the definition of BGName in Section 5.4, "Primitive Data Definitions", on page 66.		required
SubnetOrIp	A string in IP address format that represents the subnet or individual IP address.	45	Any valid IP address.	required

Table 505 Get Bg Subnet Info Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
SubnetMask	A string in IP address format that represents the subnet mask.	45	Any valid IP address.	required
Location Domain	Location Domain name	No	64	[1…64]
BOName	**V7.00.01.0:** Branch Office Name to be associated with BGSubnet	char	0…30	Optional
IpV6Address	IPV6 address. V7.00.19.01	char	Any valid IP address.	N/A
IpV6PrefixLen	IPV6 mask. V7.00.19.01	integer		N/A

Table 505 Get Bg Subnet Info Input Fields (Sheet 2 of 2)

Table 506 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
BGName	Business Group Name.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		N/A
SubnetOrIp	A string in IP address format that represents the subnet or individual IP address.	1..45	Any valid IP address.	N/A
SubnetMask	A string in IP address format that represents the subnet mask.	1..45	Any valid IP address.	N/A
ScanRate	Used by a background task to determine how often to scan a subnet for L2 changes.	integer	0..999 (seconds)	0
ScanResponseTimeout	Determines the timeout value used when waiting for responses from L2 switches.	integer	1..9999 (ms)	5
Description	Alphanumeric string describing this BG Subnet entry.	0..128	Character string.	“”
LIN	The Location Identification Number for the subnet; used for all L2 switches discovered on this subnet. Optional. If not entered, defaults to blank (in this case indicating that this Subnet/IP is to be specifically *excluded* from a LIN that covers other entries of which this Subnet/IP would normally be a part).	0..20	Numeric string.	“”
Callback Number (CBN)	The Callback Number. Mandatory.	0..25	Numeric string	N/A

Table 506 Get Bg Subnet Info Output Fields (Sheet 1 of 3)

Field Name	Description	Length	Range	Default Value
Default Callback Number (DCBN)	When PSAP is not available, the call is redirected to a default callback number. Optional.	0..25	Numeric string	N/A
Use Default Callback Number	If set to true, the default callback number is used instead of the callback number. Optional.	Boolean	false, true	false
RouteNumber	Appended to the subscriber-dialed number for retranslation and routing of the call to the PSAP operator.	0..4	Numeric string.	0
Passcode	Enables administrators to secure access to their subnet table. This is for use in a future phase when data for multiple OpenScape Voice system's is aggregated at a higher level and the aggregate table is accessed via the OpenScape Voice Assistant. The administrator will have access only to data for which the matching Passcode is supplied.	6..20	Character string.	N/A
ALIDBID	Automatic Location Identification Database ID. For use by components external to the OpenScape Voice system to organize BG Subnet data.	integer	0..9999	0
SendLINInsteadCPN	If true, indicates that the LIN is to be sent instead of the CPN (Calling Party Number) for E911 calls. Note: All of the calling subnets must be defined since endpoints can reside in different locations other than the gateway. Therefore, if the BGL's number is to be sent on a E911 Call, leave the Send LIN instead of CPN parameter as the default (false) value. However, if the BG main number (or any other number other than the BGL) is to be sent on a E911 Call, set this parameter, Send LIN instead of CPN, to true. The endpoint profile for the outbound gateway must have the full privacy service flagged.	Boolean	false, true	false
IpV6Address	IPV6 address. V7.00.19.01	char	Any valid IP address.	N/A

Table 506 Get Bg Subnet Info Output Fields (Sheet 2 of 3)

Field Name	Description	Length	Range	Default Value
IpV6PrefixLen	IPV6 mask. V7.00.19.01	integer		N/A
Location Domain	Location Domain name	No	64	[1…64]

Table 506 Get Bg Subnet Info Output Fields (Sheet 3 of 3)

11.14.2 Create Bg Subnet

This method creates a BG subnet.

Table 507 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
V5.00.18.01		subRC_APP_LOCATION_DOMAIN_LEN_INVALID subRC_APP_DSA_LOCATION_DOMAIN_NAME_SIP_PROXY_INVALID
05.03.01	N/A	subRC_SUCCESS subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_SUBNET_OR_IP_LEN_INVALID subRC_APP_BG_SUBNET_IP_INVALID subRC_APP_BG_SUBNET_MASK_LEN_INVALID subRC_APP_BG_SUBNET_MASK_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_SUBNET_PASSCODE_LEN_INVALID subRC_APP_BG_SUBNET_DESCRIPTION_LEN_INVALID subRC_APP_BG_SUBNET_LIN_LEN_INVALID subRC_APP_BG_SUBNET_ROUTE_NUMBER_LEN_INVALID subRC_APP_BG_SUBNET_SCAN_RATE_INVALID subRC_APP_BG_SUBNET_SCAN_RES_OUT_OF_RANGE subRC_APP_BG_SUBNET_ALIDBID_INVALID subRC_APP_BG_SUBNET_EXIST subRC_DB_BG_SUBNET_INSERTION_FAILED
V4.00.01.08		subRC_SUCCESS subRC_APP_BG_SUBNET_IP_INVALID subRC_APP_BG_SUBNET_MASK_INVALID subRC_APP_BG_SUBNET_SCAN_RES_OUT_OF_RANGE subRC_APP_BG_SUBNET_ALIDBID_INVALID subRC_APP_BG_SUBNET_EXIST subRC_DB_BG_SUBNET_INSERTION_FAILED subRC_BG_SUBNET_ALREADY_EXISTS subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_SUBNET_MAX_LINS_NUMBER_EXCEED subRC_APP_BG_SUBNET_LIN_INVALID subRC_APP_BG_SUBNET_CBN_INVALID subRC_APP_BG_SUBNET_DCBN_INVALID subRC_APP_BG_SUBNET_USEDCBN_INVALID subRC_APP_BG_SUBNET_LIN_LIST_ENTRIES_FAIL subRC_APP_BG_SUBNET_MEMORY_ALLOCATION_ERROR

Table 507 Create Bg Subnet Return Codes

Table 508 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
BGName	Business Group Name.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		N/A

Table 508 Create Bg Subnet Input Fields (Sheet 1 of 3)

Field Name	Description	Length	Range	Default Value
SubnetOrIp	A string in IP address format that represents the subnet or individual IP address.	1..45	Any valid IP address.	N/A
SubnetMask	A string in IP address format that represents the subnet mask.	1..45	Any valid IP address.	N/A
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
BOName	**V7.00.01.0:** Branch Office Name to be associated with BGSubnet	char	0…30	Optional
ScanRate	Used by a background task to determine how often to scan a subnet for L2 changes.	integer	0..999 (seconds)	0
ScanResponseTimeout	Determines the timeout value used when waiting for responses from L2 switches.	integer	1..9999 (ms)	5
Description	Alphanumeric string describing this BG Subnet entry.	0..128	Character string.	“”
Enter the number of LINList	Number of LIN entries to delete, If set 0 the whole BgSubnet is deleted with all of its LIN entries	6	1-100000	N/A
LIN	The Location Identification Number for the subnet; used for all L2 switches discovered on this subnet. Optional. If not entered, defaults to blank (in this case indicating that this Subnet/IP is to be specifically *excluded* from a LIN that covers other entries of which this Subnet/IP would normally be a part).	0..20	Numeric string.	“”
Callback Number (CBN)	The Callback Number. Mandatory.	0..25	Numeric string	N/A
Default Callback Number (DCBN)	When the PSAP is not available, the call is redirected to a default callback number. Optional.	0..25	Numeric string	“ “
Use Default Callback Number	If set to true, the default callback number is used instead of the callback number.	Boolean	true, false	false
AllOrNone	If set to true, no LIN entries are provisioned, even if only one entry fails. If set to false, only valid entries are provisioned.	Boolean	true, false	false

Table 508 Create Bg Subnet Input Fields (Sheet 2 of 3)

Field Name	Description	Length	Range	Default Value
RouteNumber	Appended to the subscriber-dialed number for retranslation and routing of the call to the PSAP operator.	0..4	Numeric string.	0
Passcode	Enables administrators to secure access to their subnet table. This is for use in a future phase when data for multiple 8000s is aggregated at a higher level and the aggregate table is accessed via the OpenScape Voice Assistant. The administrator will have access only to data for which the matching Passcode is supplied.	6..20	Character string.	N/A
ALIDBID	Automatic Location Identification Database ID. For use by components external to the OpenScape Voice system to organize BG Subnet data. Optional.	integer	0..9999	0
Location Domain	Location Domain name	No	64	[1…64]
IpV6Address	IPV6 address. V7.00.19.01	char	Any valid IP address.	N/A
IpV6PrefixLen	IPV6 mask. V7.00.19.01	integer		N/A

Table 508 Create Bg Subnet Input Fields (Sheet 3 of 3)

11.14.3 Get Bg Subnet List

This method retrieves and lists subnets.

Table 509 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
05.03.01	N/A	subRC_SUCCESS subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_SUBNET_OR_IP_LEN_INVALID subRC_APP_BG_SUBNET_IP_INVALID subRC_APP_BG_SUBNET_MASK_LEN_INVALID subRC_APP_BG_SUBNET_MASK_INVALID subRC_APP_BG_SUBNET_LIN_LEN_INVALID subRC_APP_BG_SUBNET_LIN_INVALID subRC_APP_BG_SUBNET_ROUTE_NUMBER_LEN_INVALID subRC_APP_BG_SUBNET_ALIDBID_INVALID subRC_DB_BG_SUBNET_LOOKUP_FAILED subRC_SYS_BG_SUBNET_ALLOC_ERR subRC_DB_BG_LOOKUP_FAILED

Table 509 Get Bg Subnet List Return Codes

Table 510 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default Value
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			
BGName	Business Group name. Optional. Provided for filtering.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		“”
SubnetOrIp	A string in IP address format that represents the subnet or individual IP address. Provided for filtering.	1..45	Any valid IP address.	N/A
SubnetMask	A string in IP address format that represents the subnet mask. Provided for filtering.	1..45	Any valid IP address.	N/A
IpV6Address	A IPV6 Address	string		
IpV6PrefixLen	IPV6 prefix length	Integer		
BOName	**V7.00.01.0:** Branch Office Name to be associated with BGSubnet	char	0…30	Optional

Table 510 Get Bg Subnet List Input Fields (Sheet 1 of 3)

Field Name	Description	Length	Range	Default Value
LIN	The Location Identification Number for the subnet; used for all L2 switches discovered on this subnet. Optional. If not entered, defaults to blank (in this case indicating that this Subnet/IP is to be specifically *excluded* from a LIN that covers other entries of which this Subnet/IP would normally be a part). Provided for filtering.	0..20	Numeric string.*	N/A
Callback Number (CBN)	The callback number. Mandatory.	0..25	Numeric string	N/A
Default Callback Number (DCBN)	When the PSAP is not available, the call is redirected to a default callback number. Optional.	0..25	Numeric string	N/A
Use Default Callback Number	If set to true, the default callback number is used instead of the callback number.	Boolean	true, false	false
RouteNumber	Appended to the subscriber-dialed number for retranslation and routing of the call to the PSAP operator. Provided for filtering.	0..4	Numeric string.	N/A
ALIDBID	Automatic Location Identification Database ID. For use by components external to the OpenScape Voice system to organize BG Subnet data. Optional. Provided for filtering.	int	0..9999	N/A
SendLINInsteadCPN	If true, indicates that the LIN is to be sent instead of the CPN (Calling Party Number) for E911 calls. Note: All of the calling subnets must be defined since endpoints can reside in different locations other than the gateway. Therefore, if the BGL's number is to be sent on a E911 Call, leave the Send LIN instead of CPN parameter as the default (false) value. However, if the BG main number (or any other number other than the BGL) is to be sent on a E911 Call, set this parameter, Send LIN instead of CPN, to true. The endpoint profile for the outbound gateway must have the full privacy service flagged.	Boolean	false, true	false

Table 510 Get Bg Subnet List Input Fields (Sheet 2 of 3)

Field Name	Description	Length	Range	Default Value
Location Domain	Location Domain name	No	64	[1…64]
GetBgSubnetList D ataOptionList	A list of extra BG Subnet fields to be retrieved	enum	BgSubnetListDataInclDescription _ (V7.00.01.06) BgSubnetListDataInclBOName (V7.00.01.09)	

Table 510 Get Bg Subnet List Input Fields (Sheet 3 of 3)

* One or more occurrences of the asterisk (*) are allowed within the LIN string on this request only to represent a wild card character. BG Subnet entries whose LIN string matches the entered string with any character in the position(s) specified by the * will be returned.

Table 511 lists the *output* fields for this method. A list of elements, each containing the following data items, is returned.

Field Name	Description	Length	Range	Default Value
BGName	Business Group Name.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		N/A
SubnetOrIp	A string in IP address format that represents the subnet or individual IP address.	1..45	Any valid IP address.	N/A
SubnetMask	A string in IP address format that represents the subnet mask.	1..45	Any valid IP address.	N/A
BOName	Branch Office Name to be associated with BGSubnet V7.00.01.09	char	0…30	Optional
Description	Alphanumeric string describing this BG Subnet entry. V7.00.01.06	0..128	Character string.	“”

Table 511 Get Bg Subnet List Output Fields

11.14.4 Delete Bg Subnet

This method deletes BG Subnets from the OpenScape Voice system.

Table 512 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
V5.00.18.01		subRC_APP_LOCATION_DOMAIN_LEN_INVALID subRC_APP_DSA_LOCATION_DOMAIN_NAME_SIP_PROXY_INVALID

Table 512 Delete Bg Subnet Return Codes (Sheet 1 of 2)

WSDL Version	Obsolete in WSDL Version	Return Codes
05.03.01	N/A	subRC_SUCCESS subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_SUBNET_OR_IP_LEN_INVALID subRC_APP_BG_SUBNET_IP_INVALID subRC_APP_BG_SUBNET_MASK_LEN_INVALID subRC_APP_BG_SUBNET_MASK_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_DB_BG_SUBNET_DELETION_FAILURE
V4.00.01.08		subRC_SUCCESS subRC_APP_BG_SUBNET_IP_INVALID subRC_APP_BG_SUBNET_MASK_INVALID subRC_DB_BG_SUBNET_DELETION_FAILURE subRC_DB_BG_SUBNET_LOOKUP_FAILED subRC_APP_BG_SUBNET_MEMORY_ALLOCATION_ERROR

Table 512 Delete Bg Subnet Return Codes (Sheet 2 of 2)

Table 513 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
BGName	Business Group Name.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		required
SubnetOrIp	A string in IP address format that represents the subnet or individual IP address. Mandatory for this request.	1..45	Any valid IP address.	N/A
SubnetMask	A string in IP address format that represents the subnet mask. Mandatory for this request.	1..45	Any valid IP address.	N/A
Enter the number of LINList	Number of LIN entries to delete, If set to 0, the entire BgSubnet is deleted with all of its LIN entries	6	1-100000	N/A
Location Domain	Location Domain name	No	64	[1…64]
IpV6Address	IPV6 address. V7.00.19.01	char	Any valid IP address.	N/A
IpV6PrefixLen	IPV6 mask. V7.00.19.01	integer		N/A

Table 513 Delete Bg Subnet Input Fields

11.14.5 Update Bg Subnet

This method modifies a BG subnet

Table 514 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
V5.00.18.01		subRC_APP_LOCATION_DOMAIN_LEN_INVALID subRC_APP_DSA_LOCATION_DOMAIN_NAME_SIP_PROXY_INVALID
05.03.01	N/A	subRC_SUCCESS subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_SUBNET_OR_IP_LEN_INVALID subRC_APP_BG_SUBNET_IP_INVALID subRC_APP_BG_SUBNET_MASK_LEN_INVALID subRC_APP_BG_SUBNET_MASK_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_SUBNET_PASSCODE_LEN_INVALID subRC_APP_BG_SUBNET_DESCRIPTION_LEN_INVALID subRC_APP_BG_SUBNET_LIN_LEN_INVALID subRC_APP_BG_SUBNET_ROUTE_NUMBER_LEN_INVALID subRC_APP_BG_SUBNET_SCAN_RATE_INVALID subRC_APP_BG_SUBNET_SCAN_RES_OUT_OF_RANGE subRC_APP_BG_SUBNET_ALIDBID_INVALID subRC_APP_BG_SUBNET_EXIST subRC_DB_BG_SUBNET_INSERTION_FAILED
V4.00.01.08		subRC_SUCCESS subRC_APP_BG_SUBNET_IP_INVALID subRC_APP_BG_SUBNET_MASK_INVALID subRC_APP_BG_SUBNET_DOES_NOT_EXIST subRC_DB_BG_SUBNET_LOOKUP_FAILED subRC_APP_BG_SUBNET_SCAN_RATE_INVALID subRC_APP_BG_SUBNET_SCAN_RES_OUT_OF_RANGE subRC_DB_BG_SUBNET_MODIFY_FAILED subRC_APP_BG_SUBNET_MAX_LINS_NUMBER_EXCEED subRC_APP_BG_SUBNET_MEMORY_ALLOCATION_ERROR subRC_APP_BG_SUBNET_LIN_LIST_ENTRIES_FAIL

Table 514 Update Bg Subnet Return Codes

Table 515 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
BGName	Business Group Name.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		required
SubnetOrIp	A string in IP address format that represents the subnet or individual IP address. Mandatory for this request.	1..45	Any valid IP address.	N/A
SubnetMask	A string in IP address format that represents the subnet mask. Mandatory for this request.	1..45	Any valid IP address.	N/A
BOName	Branch Office Name to be associated with BGSubnet V7.00.01.09	char	0…30	Optional

Table 515 Update Bg Subnet Input Fields (Sheet 1 of 4)

Field Name	Description	Length	Range	Default Value
IpV6Address	IPV6 address. V7.00.19.01	char	Any valid IP address.	N/A
IpV6PrefixLen	IPV6 mask. V7.00.19.01	integer		N/A
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
ScanRate	Used by a background task to determine how often to scan a subnet for L2 changes. Optional.	integer	0..999 (seconds)	0
ScanResponseTimeout	Determines the timeout value used when waiting for responses from L2 switches.	integer	1..9999 (ms)	5
Description	Alphanumeric string describing this BG Subnet entry.	0..128	Character string.	“““
Enter the number of LINList	Number of LIN entries to delete, If set to 0, the entire BgSubnet is deleted with all of its LIN entries	6	1-100000	N/A
LIN	The Location Identification Number for the subnet; used for all L2 switches discovered on this subnet. Optional. If not entered, defaults to blank (in this case indicating that this Subnet/IP is to be specifically *excluded* from a LIN that covers other entries of which this Subnet/IP would normally be a part). Provided for filtering.	0..20	Numeric string.	“ “
Callback Number (CBN)	The callback number. Mandatory.	0..25	Numeric string	N/A
Default Callback Number (DCBN)	When the PSAP is not available, the call is redirected to a default callback number. Optional.	0..25	Numeric string	N/A
Use Default Callback Number	If set to true, the default callback number is used instead of the callback number.	Boolean	true, false	N/A
AllOrNone	If set to true, no LIN entries are provisioned, even if only one entry fails. If set to false, only valid entries are provisioned.	Boolean	true, false	false
RouteNumber	Appended to the subscriber-dialed number for retranslation and routing of the call to the PSAP operator. Optional. Defaults to 0.	0..4	Numeric string.	0

Table 515 Update Bg Subnet Input Fields (Sheet 2 of 4)

Field Name	Description	Length	Range	Default Value
Passcode	Enables administrators to secure access to their subnet table. This is for use in a future phase when data for multiple OpenScape Voice system's is aggregated at a higher level and the aggregate table is accessed via the OpenScape Voice Assistant. The administrator will have access only to data for which the matching Passcode is supplied. Mandatory.	6..20	Character string.	N/A
ALIDBID	Automatic Location Identification Database ID. For use by components external to the OpenScape Voice system to organize BG Subnet data. Optional.	integer	0..9999	0
NewSubnetOrIp	Update is for the subnet or IP	string		
NewSubnetMask	New Netmask	string		
NewIpV6Address	New IPv6 address	string		
NewIpV6PrefixLen	New IPv6 prefix	integer		
NewLocationDomain	New location domain.	string		
AllOrNone	If set to true, no LIN entries are provisioned, even if only one entry fails. If set to false, only valid entries are provisioned.	Boolean		

Table 515 Update Bg Subnet Input Fields (Sheet 3 of 4)

Field Name	Description	Length	Range	Default Value
SendLINInsteadCPN	If true, indicates that the LIN is to be sent instead of the CPN (Calling Party Number) for E911 calls. Note: All of the calling subnets must be defined since endpoints can reside in different locations other than the gateway. Therefore, if the BGL's number is to be sent on a E911 Call, leave the Send LIN instead of CPN parameter as the default (false) value. However, if the BG main number (or any other number other than the BGL) is to be sent on a E911 Call, set this parameter, Send LIN instead of CPN, to true. The endpoint profile for the outbound gateway must have the full privacy service flagged.	Boolean	false, true	false

Table 515 Update Bg Subnet Input Fields (Sheet 4 of 4)

11.14.6 Get Bg Subnet And LIN List

This method retrieves and lists subnets and their Location Identity Numbers.

Table 509 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
05.03.01	N/A	subRC_SUCCESS subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_SUBNET_OR_IP_LEN_INVALID subRC_APP_BG_SUBNET_IP_INVALID subRC_APP_BG_SUBNET_MASK_LEN_INVALID subRC_APP_BG_SUBNET_MASK_INVALID subRC_APP_BG_SUBNET_LIN_LEN_INVALID subRC_APP_BG_SUBNET_LIN_INVALID subRC_APP_BG_SUBNET_ROUTE_NUMBER_LEN_INVALID subRC_APP_BG_SUBNET_ALIDBID_INVALID subRC_DB_BG_SUBNET_LOOKUP_FAILED subRC_SYS_BG_SUBNET_ALLOC_ERR subRC_DB_BG_LOOKUP_FAILED

Table 516 Get Bg Subnet List Return Codes

Table 510 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default Value
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			
OperatorId				
BGName	Business Group name. Optional. Provided for filtering.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		“”
BOName	**V7.00.01.0:** Branch Office Name to be associated with BGSubnet	char	0…30	Optional
LastSyncTime	Time when last synched		dateTime	
ScanRate	Used by a background task to determine how often to scan a subnet for L2 changes. Optional.	integer	0..999 (seconds)	0
ScanResponseTimeout	Determines the timeout value used when waiting for responses from L2 switches.	integer	1..9999 (ms)	5
Description	Alphanumeric string describing this BG Subnet entry.	0..128	Character string.	“”
Number of LINList	Number of LIN entries to get,	6	1-100000	N/A
LIN	The Location Identification Number for the subnet; used for all L2 switches discovered on this subnet. Optional. If not entered, defaults to blank (in this case indicating that this Subnet/IP is to be specifically *excluded* from a LIN that covers other entries of which this Subnet/IP would normally be a part). Provided for filtering.	0..20	Numeric string.	“ “
AllOrNone	If set to true, no LIN entries are provisioned, even if only one entry fails. If set to false, only valid entries are provisioned.	Boolean	true, false	false
RouteNumber	Appended to the subscriber-dialed number for retranslation and routing of the call to the PSAP operator. Optional. Defaults to 0.	0..4	Numeric string.	0

Table 517 Get Bg Subnet and LIN List Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
Passcode	Enables administrators to secure access to their subnet table. This is for use in a future phase when data for multiple OpenScape Voice system's is aggregated at a higher level and the aggregate table is accessed via the OpenScape Voice Assistant. The administrator will have access only to data for which the matching Passcode is supplied. Mandatory.	6..20	Character string.	N/A
ALIDBID	Automatic Location Identification Database ID. For use by components external to the OpenScape Voice system to organize BG Subnet data. Optional.	integer	0..9999	0
SendLINInsteadCPN		Bool		
AssociatedEndpointIP		string		
RepresentativeEndpointName		string		

Table 517 Get Bg Subnet and LIN List Input Fields (Sheet 2 of 2)

Table 511 lists the *output* fields for this method. A list of elements, each containing the following data items, is returned.

Field Name	Description	Length	Range	Default Value
BGName	Business Group Name.	Please refer to the definition of BGName in Section 5.4, "Primitive Data Definitions", on page 66.		N/A
SubnetOrIp	A string in IP address format that represents the subnet or individual IP address.	1..45	Any valid IP address.	N/A
SubnetMask	A string in IP address format that represents the subnet mask.	1..45	Any valid IP address.	N/A
BOName	Branch Office Name to be associated with BGSubnet V7.00.01.09	char	0…30	Optional
Description	Alphanumeric string describing this BG Subnet entry. V7.00.01.06	0..128	Character string.	""

Table 518 Get Bg Subnet List Output Fields

11.15 Business Group Authorization Codes

Depending on the traffic type, an off-network authorization code may be required to allow external calls. The authorization codes are defined at the BG level, not at the BGL level. Each BG can have 50,000 authorization codes defined. The maximum number of authorization codes that can be defined on the switch is 100,000. Each time a BG subscriber dials a number that corresponds to an external call, an authorization code may be provided. If provided, it is validated by call processing to determine whether the subscriber is authorized to make the call. With the SOAP requests described in the following section, the SOAP Server provides the ability to create, delete, and retrieve BG authorization codes. All of these requests can be done for one authorization code or a list of authorization codes.

11.15.1 Create Auth Code

This method creates authorization codes. It supports the association of an Authorization Code with a Subcriber DN and a type (***business*** or ***private***). If the crafts person enters a Subscriber DN then he/she shall also have to select a type and vice versa.

Table 519 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01.17		subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_AUTH_CODE_LEN_INVALID subRC_APP_BG_AUTH_CODE_MISSING subRC_APP_BG_AUTH_CODE_INVALID subRC_APP_BG_AUTH_CODE_SYS_OUT_OF_RANGE subRC_APP_BG_AUTH_CODE_OUT_OF_RANGE subRC_APP_BG_AUTH_CODE_EXIST subRC_DB_BG_AUTH_CODE_LOOKUP_FAILED subRC_DB_BG_AUTH_CODE_INSERTION_FAILED subRC_WAR_BG_AUTH_CREATION_FAILED subRC_SUCCESS

Table 519 Create Auth Code Return Codes

Table 520 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			

Table 520 Create Auth Code Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
BGName	Business Group Name.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		required
*__ptrAuth CodeEntry	Pointer to the first auth code of the list to be created.*	N/A	N/A	N/A
__size	Number of the elements in the list of auth codes to be created.*	int	1..50000	N/A
*Member of the unsp_AuthCodeList.				
ServiceId	Service DN	Char		N/A
AuthCodeType	The type of the Authorization	Enum	AuthCodeNone, AuthCodeBusiness, AuthCodePrivate	N/A

Table 520 Create Auth Code Input Fields (Sheet 2 of 2)

11.15.2 Delete Auth Code

This method deletes an authorization code.

Table 521 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01.17		subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_AUTH_CODE_NUMBER_DEL_INVALID subRC_APP_BG_AUTH_CODE_INVALID subRC_APP_BG_AUTH_CODE_LEN_INVALID subRC_APP_BG_AUTH_CODE_DOES_NOT_EXIST subRC_DB_BG_AUTH_CODE_LOOKUP_FAILED subRC_APP_BG_AUTH_CODE_DELETION_FAILED subRC_WAR_BG_AUTH_DELETION_FAILED subRC_SUCCESS

Table 521 DeleteAuthCode Return Codes

Table 522 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			

Table 522 Delete Auth Code Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
BGName	Business Group Name.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		required
*__ptrAuth CodeEntry	Pointer to the first auth code of the list to be created.*	N/A	N/A	N/A
__size	Number of the elements in the list of auth codes to be created.*	int	1..50000	N/A
*Member of the unsp_AuthCodeList.				

Table 522 Delete Auth Code Input Fields (Sheet 2 of 2)

11.15.3 Get Auth Code List

This method supports the retrieval of the list of Authorization Codes with their associated SubscriberDN and type. Paging, sorting (Ascending/Descending) and filtering on SubscriberDN and Type.

Table 523 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01.17		subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_AUTH_CODE_OUT_OF_RANGE subRC_SYS_BG_AUTH_CODE_ALLOC_ERR subRC_DB_BG_AUTH_CODE_LOOKUP_FAILED subRC_APP_BG_AUTH_CODE_DOES_NOT_EXIST subRC_SUCCESS

Table 523 Get Auth Code List Return Codes

Table 524 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
BGName	Business Group Name.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		required
ServiceId	Service DN	Char		N/A

Table 524 Get Auth Code List Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
AuthCodeType	The type of the Authorization	Enum	Undefined, Business, Private	N/A
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			

Table 524 Get Auth Code List Input Fields (Sheet 2 of 2)

11.15.3.1 Get Auth Code

This method retrieves an authorization code.

Table 525 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01.17		subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_AUTH_CODE_MISSING subRC_APP_BG_AUTH_CODE_LEN_INVALID subRC_DB_BG_AUTH_CODE_LOOKUP_FAILED subRC_APP_BG_AUTH_CODE_DOES_NOT_EXIST subRC_SUCCESS

Table 525 Get Auth Code Return Codes

Table 526 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
BGName	Business Group Name.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		required
AuthCode	Authorization Code. Refer to Section 5.4, “Primitive Data Definitions”, on page 66.			

Table 526 Get Auth Code Input Fields

11.16 Modify Auth Code

Interface for supporting the update of an Authorization Code and/or the associated SubscriberDN and/or the Type. If the crafts person enters a Subscriber DN then he/she shall also have to select a type and vice versa. Validations shall be performed by both RSS/SOAP.

Table 527 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)
BGName	Business Group Name to be associated with Authorization Code	Char	1…30	mandatory
AuthCode	Authorization Code to be modified	Char	2…14	mandatory
NewAuthCode	New Authrization Code	Char	2…14	optional
ServiceId	Service DN	Char		N/A
AuthCodeType	The type of the Authorization	Enum	Undefined, Business, Private	N/A

Table 527 Get Auth Code Input Fields

11.17 Private Numbering Plan

A Private Numbering Plan may be used to define allowable dialing, translation, and routing for a Business Group. The Numbering Plan Name is provisioned as part of the Business Group data. If no Numbering Plan ID/Numbering Plan Name are provided, the default is provisioned as the system default Private Numbering Plan (1).

The specified Business Group Name is provisioned internally as the Customer Name. The Numbering Plan information cannot be modified.

The provisioning of Private Numbering Plans for Business Groups is accomplished through the OpenScape Voice Assistant. The system default Numbering Plan can also be used to assign to a Business Group.

The Number Plan is defined when the BG is created and may not be changed. If not specified on BG creation, the system default will apply.

The system defaults are provisioned as follows:

- PSTN Switch: NumberPlanId: 9999 NumberPlanNAme: E164DEFAULT

  NumberPlanType: NPTypeE164NANP

Only the system default Numbering Plan can be shared among different BGs. A Private Numbering Plan associated with a BG cannot be shared by another BG.

For more information about Numbering Plans, refer to Chapter 12, “Numbering Plans”.

11.18 Business Group System Speed Dial Lists

The SOAP requests in this section allow a Business Group administrator to define, populate, and manage BG Speed Dial lists. Out of a system pool of 10,000 BG System Speed Dial lists and 1,000,000 Speed Dial entries, a BG can be assigned up to 10,000 (1..10,000) BG System Speed Dial lists, where a list may have up to 1,000 entries (0..999). When the available pool of lists or of entries is exhausted, the BG administrator is informed during provisioning.

The lists are then usable by Business Group Lines who have the CfBgSpDial feature (see Section 11.17, "Business Group System Speed Dial Lists") subscribed and active. Each user has access to up to two of the 10,000 BG level lists via separately defined Prefix Access Codes. The user dials the appropriate Prefix Access Code followed by an Entry Id to access a particular entry on a list, and the number stored for that entry is dialed, whether it consists of a destination, an account code, etc. The Speed Dial list is available anytime the user can dial, and not limited to idle, talk, conference and all consultation dial states.

Attention: This feature is separate from, and not to be confused with the Speed Calling feature (see Section 10.46, “System Speed Dial (CfBgSpDial)”) that is also available to subscribers and BGLs.

A list entry may contain access codes for account codes, the account code itself, feature access codes, and the dialed party number (private or public). Each list entry may be up to 30 characters in length, consisting of digits, # and *. Each list has its own entries, they are not shared.

A Speed Dial list is accessible anytime a user attempts to dial another destination. The Speed Dial list entry is processed until the sequence is completed or an error condition is reached.

11.18.1 Create Bg Speed Dial List

This request creates a BG Speed Dial List.

Table 528 lists the return codes for this request.

WSDL Version	Obsolete in WSDL Version	Return Codes
11.00.01.22		subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_FEAT_BGSPDIAL_NO_ENTRY_LIST_PASSED subRC_APP_FEAT_BGSPDIAL_LIST_NAME_LEN_INVALID subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_APP_FEAT_BGSPDIAL_LIST_ID_MISSING subRC_APP_FEAT_BGSPDIAL_LIST_ID_INVALID subRC_APP_FEAT_BGSPDIAL_LIST_ID_ALREADY_EXISTS subRC_APP_FEAT_BGSPDIAL_LIST_NAME_ALREADY_EXISTS subRC_APP_FEAT_BGSPDIAL_LIST_CREATE_FAILED subRC_APP_FEAT_BGSPDIAL_ENTRY_MAX_NUMBER_EXCEED subRC_APP_FEAT_BGSPDIAL_ENTRY_ID_MISSING subRC_APP_FEAT_BGSPDIAL_ENTRY_ID_INVALID subRC_APP_FEAT_BGSPDIAL_ENTRY_DATA_LEN_INVALID subRC_APP_FEAT_BGSPDIAL_ENTRY_DATA_MISSING subRC_APP_FEAT_BGSPDIAL_ENTRY_DATA_INVALID subRC_APP_FEAT_BGSPDIAL_ENTRY_CREATE_FAILED subRC_APP_FEAT_BGSPDIAL_ENTRIES_CREATE_COMPLETE_WITH_ERRORS

Table 528 Create Bg Speed Dial List Return Codes

Table 529 lists the input fields for this request.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
BGName	Business Group Name.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		required
ListName	A name to associate with the BG Speed Dial List. This attribute is optional, and is not used as a key to identify the list; it is for informational purposes only. However, it must be unique within the BG.	0..30	Character String	“”

Table 529 Create Bg Speed Dial List Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
AllOrNone	A flag to indicate whether atomic processing is required or partial success is allowed. If True, when the processing of one entry fails, the data is rolled back to its state prior to the request. If False, when the processing of one entry fails, a best effort attempt will be made, and the subsequent entries will be processed. Note: If AllOrNone is False, and errors occur, a list of errors is returned identifying which entries could not be processed successfully.	Boolean	True, False	False
BgSpeedDialListCreate	A structure containing the following attributes:			
BgSpeedDialListCreate. ListId	The numeric identifier of the list to be created.	integer	1..10,000	mandatory
BgSpeedDialListCreate. BgSpeedDialListOfEntries		An optional list of entries to assign to the list at creation. Each entry contains the following attributes:		
BgSpeedDialListCreate. BgSpeedDialListOfEntries. EntryId	The numeric identifier of the entry.	integer	0..999	N/A
BgSpeedDialListCreate. BgSpeedDialListOfEntries. EntryData	The content of the entry, which may consist of digits, * or #.	1..30	Character String	N/A

Table 529 Create Bg Speed Dial List Input Fields (Sheet 2 of 2)

11.18.2 Update Bg Speed Dial List

This request updates the name of a BG Speed Dial List.

Table 530 lists the return codes for this request.

WSDL Version	Obsolete in WSDL Version	Return Codes
11.00.01.22		subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_FEAT_BGSPDIAL_LIST_NAME_LEN_INVALID subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_APP_FEAT_BGSPDIAL_LIST_ID_MISSING subRC_APP_FEAT_BGSPDIAL_LIST_ID_INVALID subRC_APP_FEAT_BGSPDIAL_LIST_ID_NOT_EXISTS subRC_APP_FEAT_BGSPDIAL_LIST_NAME_ALREADY_EXISTS

Table 530 Update Bg Speed Dial List Return Codes

Table 531 lists the input fields for this request.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
BGName	Business Group Name.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		required
ListId	The numeric identifier of the list to be updated.	integer	1..10,000	mandatory
ListName	A name to associate with the BG Speed Dial List. This attribute is optional, and is not used as a key to identify the list; it is for informational purposes only. However, it must be unique within the BG. To remove a ListName, the empty string may be provided.	0..30	Character String	N/A

Table 531 Update Bg Speed Dial List Input Fields

11.18.3 Update Bg Speed Dial List Entries

This request adds entries to a BG Speed Dial List or updates existing entries.

Table 532 lists the return codes for this request.

WSDL Version	Obsolete in WSDL Version	Return Codes
11.00.01.22		subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_FEAT_BGSPDIAL_NO_ENTRY_LIST_PASSED subRC_APP_FEAT_BGSPDIAL_LIST_NAME_LEN_INVALID subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_APP_FEAT_BGSPDIAL_LIST_ID_MISSING subRC_APP_FEAT_BGSPDIAL_LIST_ID_INVALID subRC_APP_FEAT_BGSPDIAL_LIST_ID_NOT_EXISTS subRC_APP_FEAT_BGSPDIAL_ENTRY_MAX_NUMBER_EXCEED subRC_APP_FEAT_BGSPDIAL_ENTRY_ID_MISSING subRC_APP_FEAT_BGSPDIAL_ENTRY_ID_INVALID subRC_APP_FEAT_BGSPDIAL_ENTRY_DATA_LEN_INVALID subRC_APP_FEAT_BGSPDIAL_ENTRY_DATA_MISSING subRC_APP_FEAT_BGSPDIAL_ENTRY_DATA_INVALID subRC_APP_FEAT_BGSPDIAL_ENTRY_CREATE_FAILED subRC_APP_FEAT_BGSPDIAL_ENTRY_UPDATE_FAILED subRC_APP_FEAT_BGSPDIAL_ENTRIES_UPDATE_COMPLETE_WITH_ERRORS

Table 532 Update Bg Speed Dial List Entries Return Codes

Table 533 lists the input fields for this request.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
BGName	Business Group Name.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		required
AllOrNone	A flag to indicate whether atomic processing is required or partial success is allowed. If True, when the processing of one entry fails, the data is rolled back to its state prior to the request. If False, when the processing of one entry fails, a best effort attempt will be made, and the subsequent entries will be processed. Note: If AllOrNone is False, and errors occur, a list of errors is returned identifying which entries could not be processed successfully.	Boolean	True, False	False
BgSpeedDialListUpdate	A structure containing the following attributes:			

Table 533 Update Bg Speed Dial List Entries Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
BgSpeedDialListUpdate. ListId	The numeric identifier of the list to be updated.	integer	1..10,000	mandatory
BgSpeedDialListUpdate. BgSpeedDialListOfEntries		The list of entries to be added to the list or to be updated in the list if they exist already. Each entry contains the following attributes:		
BgSpeedDialListUpdate. BgSpeedDialListOfEntries. EntryId	The numeric identifier of the entry.	integer	0..999	N/A
BgSpeedDialListUpdate. BgSpeedDialListOfEntries. EntryData	The content of the entry, which may consist of digits, * or #.	1..30	Character String	N/A

Table 533 Update Bg Speed Dial List Entries Input Fields (Sheet 2 of 2)

11.18.4 Delete Bg Speed Dial List Entries

This request deletes entries from a BG Speed Dial List.

Table 534 lists the return codes for this request.

WSDL Version	Obsolete in WSDL Version	Return Codes
11.00.01.22		subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_FEAT_BGSPDIAL_NO_ENTRY_LIST_PASSED subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_APP_FEAT_BGSPDIAL_LIST_ID_MISSING subRC_APP_FEAT_BGSPDIAL_LIST_ID_INVALID subRC_APP_FEAT_BGSPDIAL_LIST_ID_NOT_EXISTS subRC_APP_FEAT_BGSPDIAL_ENTRY_MAX_NUMBER_EXCEED subRC_APP_FEAT_BGSPDIAL_ENTRY_ID_MISSING subRC_APP_FEAT_BGSPDIAL_ENTRY_ID_INVALID subRC_APP_FEAT_BGSPDIAL_ENTRY_ID_NOT_EXISTS subRC_APP_FEAT_BGSPDIAL_ENTRY_REMOVE_FAILED subRC_APP_FEAT_BGSPDIAL_ENTRIES_DELETE_COMPLETE_WITH_ERRORS subRC_APP_FEAT_BGSPDIAL_DB_LOOKUP_ERROR

Table 534 Delete Bg Speed Dial List Entries Return Codes

Table 535 lists the input fields for this request.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
BGName	Business Group Name.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		required
AllOrNone	A flag to indicate whether atomic processing is required or partial success is allowed. If True, when the processing of one entry fails, the data is rolled back to its state prior to the request. If False, when the processing of one entry fails, a best effort attempt will be made, and the subsequent entries will be processed. Note: If AllOrNone is False, and errors occur, a list of errors is returned identifying which entries could not be processed successfully.	Boolean	True, False	False
BgSpeedDialListDelete	A structure containing the following attributes:			
BgSpeedDialListDelete. ListId	The numeric identifier of the list to be updated.	integer	1..10,000	mandatory
BgSpeedDialListDelete. BgSpeedDialListOfEntries		The list of entries to be deleted from the list. Each entry contains the following attributes:		
BgSpeedDialListDelete. BgSpeedDialListOfEntries. EntryId	The numeric identifier of the entry.	integer	0..999	N/A

Table 535 Delete Bg Speed Dial List Entries Input Fields

11.18.5 Delete Bg Speed Dial List

This request deletes an entire BG Speed Dial List including all its entries.

Table 536 lists the return codes for this request.

WSDL Version	Obsolete in WSDL Version	Return Codes
11.00.01.22		subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_APP_FEAT_BGSPDIAL_LIST_ID_MISSING subRC_APP_FEAT_BGSPDIAL_LIST_ID_INVALID subRC_APP_FEAT_BGSPDIAL_LIST_ID_NOT_EXISTS subRC_APP_FEAT_BGSPDIAL_ENTRY_NUMBER_RETRIEVE_ERROR subRC_APP_FEAT_BGSPDIAL_ENTRY_NUMBER_RETRIEVE_ERROR subRC_APP_FEAT_BGSPDIAL_LIST_REMOVE_FALIED

Table 536 Delete Bg Speed Dial List Return Codes

Table 537 lists the input fields for this request.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
BGName	Business Group Name.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		required
ListId	The numeric identifier of the list to be updated.	integer	1..10,000	mandatory

Table 537 Delete Bg Speed Dial List Input Fields

11.18.6 Get Bg Speed Dial List

This request retrieves a BG Speed Dial List.

Table 538 lists the return codes for this request.

WSDL Version	Obsolete in WSDL Version	Return Codes
11.00.01.22		subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_APP_FEAT_BGSPDIAL_LIST_ID_MISSING subRC_APP_FEAT_BGSPDIAL_LIST_ID_INVALID subRC_APP_FEAT_BGSPDIAL_STARTING_INDEX_INVALID subRC_APP_FEAT_BGSPDIAL_LIST_ID_NOT_EXISTS subRC_APP_FEAT_BGSPDIAL_ENTRY_NUMBER_RETRIEVE_ERROR

Table 538 Get Bg Speed Dial List Return Codes

Table 539 lists the input fields for this request.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.			
InterfaceWSDL	The WSDL version of the request.	Refer to Section 5.1, "Interface (WSDL) Versions", on page 61.	Character string	Latest
BGName	Business Group Name.	Please refer to the definition of BGName in Section 5.4, "Primitive Data Definitions", on page 66.		required
ListId	The numeric identifier of the list to be updated.	integer	1..10,000	mandatory
StartingIndex	The entry ID of the entry with which to start. By default, the starting index is the beginning of the list.	integer	0..999	0
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, "Paging", on page 72.			
SortType	Provides sorting on EntryId and on EntryData	Enumerator	SortByEntryId, SortByEntryData	SortByEntryId
SortOrder	Provides selection of sorting order	Enumerator	SortOrderAscending, SortOrderDescending	SortOrderAscending
FilterType	Provides filtering on EntryId and on EntryData	Enumerator	FilterBgSpeedListEntryId, FilterBgSpeedListEntryData	FilterBgSpeedListEntryId

Table 539 Get Bg Speed Dial List Input Fields

Table 540 lists the output fields for this request.

Field Name	Description	Length	Range	Default Value
BGName	Business Group Name.	Please refer to the definition of BGName in Section 5.4, "Primitive Data Definitions", on page 66.		required
ListName	A name to associate with the BG Speed Dial List. This attribute is optional, and is not used as a key to identify the list; it is for informational purposes only. However, it must be unique within the BG. To remove a ListName, the empty string may be provided.	0..30	Character String	N/A

Table 540 Get Bg Speed Dial List Output Fields

Field Name	Description	Length	Range	Default Value
BgSpeedDialList	A structure containing the following attributes:			
BgSpeedDialList. ListId	The numeric identifier of the list.	integer	1..10	N/A
BgSpeedDialList. BgSpeedDialListOfEntries		The list of entries in the list. Each entry contains the following attributes:		
BgSpeedDialList. BgSpeedDialListOfEntries. EntryId	The numeric identifier of the entry.	integer	0..999	N/A

Table 540 Get Bg Speed Dial List Output Fields

11.18.7 Get Bg Speed Dial List Ids

This request retrieves BG Speed Dial List Ids.

Table 541 lists the return codes for this request.

WSDL Version	Obsolete in WSDL Version	Return Codes
V5.00.38.01		subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_DB_FEAT_BGSPDIAL_DB_LOOKUP_ERROR

Table 541 Get Bg Speed Dial List Ids Return Codes

Table 542 lists the input fields for this request.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 80.			
InterfaceWSDL	The WSDL version of the request.	Refer to Section 5.1, “Interface (WSDL) Versions”, on page 75.		Character string
BGName	Business Group Name.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 80.		
ListId	The numeric identifier of the list to be updated.	integer	1..10,000	
ListName	The name of the Speed Dial List.	Character String	0..30	

Table 542 Get Bg Speed Dial List Ids Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 88.			
SortType	Provides sorting on List Id, List Name, Number of Entries	Enumerator	SortByListId, SortByListName, SortByNumberOfEntries	
SortOrder	Provides selection of sorting order	Enumerator	SortOrderAscending, SortOrderDescending	
FilterType	Provides filtering on List Id and List Name	Enumerator	FilterBgSpeedListIds, FilterBgSpeedListIdsName	

Table 542 Get Bg Speed Dial List Ids Input Fields (Sheet 2 of 2)

Table 543 lists the output fields for this request.

Field Name	Description	Length	Range	Default Value
BGName	Business Group Name.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 80.		required
BgSpeedDialListIds .ListId	The numeric identifier of the list.	Integer	1..10,000	N/A
BgSpeedDialListIds .ListName	The name of the Speed Dial List	0..30	Character String	N/A
BgSpeedDialListIds .NumberOfEntries	The number of entries of the Speed Dial List	Integer	0..999	N/A
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below. Result			
PaginatedInfo	A common paging structure containing Current Page/ Num of Pages / Page Size etc. PaginatedInfo			
BgSpeedDialListIds	A List containing the List ID, List Name and Number of Entries for each List ID			

Table 543 Get Bg Speed Dial List Ids Output Fields

11.19 Business Group Parking Lots

11.19.1 Create Parking Lot

Unsp__CreateParkingLot creates parking lot entries inside a business group. It is placed in the Business Group menu (400) in soapCli under the general ParkingLot menu (480). The administrator can provision a parking lot through a single Unsp__CreateParkingLot request. The SOAP server returns success or failure along with the error message describing the reason for the failure.

Table 544 lists the return codes for this request.

Unsp__CreateParkingLotResult		
Attribute Name	**Values**	**Messages**
Unsp__ResultCodeStruct	Success	Parking Lot <name> created Successfully
	Failure	No Parking Lot was created • Invalid BG name • Invalid Parking Lot name • Parking Lot already exists • Invalid Recall timer Value. Allowed values 30-600 with stepping of 30 • Invalid ParkSpaceStart value. Value length must not exceed 5 digits. • Invalid ParkSpaceEnd value. Value length must not exceed 5 digits. • Limit of 10000 Parking Spaces per system is exceeded.

Table 544 Create Parking Lot Result Fields

Table 545 lists the input fields for this request.

Unsp__CreateParkingLotRequest			
Attribute Name	**Type**	**Default Value**	**Description**
BGName	unsp__BGName	NA	Mandatory. The BG where the parking lot is assigned. Length is 30 chars maximum.
ParkingLotName	xsd__string	NA	Mandatory. The parking lot name. Length is 20 chars maximum.
RecallTimer	int	180	Optional. A value in steps of 30. Minimum value: 30. Maximum value: 600
ParkSpaceStart	int	0	Optional. 5 digits long at maximum. When not set starting space is at position 0
ParkSpaceEnd	int	9	Optional. 5 digits long at maximum. When not set, ending space is at position 9

Table 545 Create Parking Lot Request Fields

11.19.2 Update Parking Lot

unsp__UpdateParkingLot updates existing parking lot entities inside a business group, with one entry per request. It is placed in the Business Group menu (400) in soapCli under the general ParkingLot menu (480). New parking lots cannot be provisioned from this interface. Furthermore, there is no capability to move a parking lot to a different business group. The administrator can modify all values of the request structure except for the BG name. Unsp__UpdateParkingLot is used to modify the values of the following fields:

- RecallTimer
- ParkSpaceStart
- ParkSpaceEnd
- ParkingLotName

Note: When the administrator modifies the parking lot name, all parked calls are lost. SOAP will proceed with the update and issues a message that will inform about the loss of parked calls.

Table 546 lists the return codes for this request.

Unsp__UpdateParkingLotResult		
Attribute Name	**Values**	**Messages**
Unsp__ResultCodeStruct	Success	Parking Lot(s) updated Successfully
	Failure	No Parking Lot was updated • Invalid BG name • Invalid Parking Lot name • Parking Lot does not exist • Invalid Recall timer Value. Allowed values 30-600 with stepping of 30 • Invalid ParkSpaceStart value. Value length must not exceed 5 digits. • Invalid ParkSpaceEnd value. Value length must not exceed 5 digits. • Limit of 10000 Parking Spaces per system is exceeded(Req[3008]) • Proposed name for ParkingLotName already exists. Please use a different ParkingLotName or delete the existing one.

Table 546 Create Parking Lot Result Fields

Table 547 lists the input fields for this request.

Unsp__UpdateParkingLotRequest			
Attribute Name	**Type**	**Default Value**	**Description**
BGName	unsp__BGName	NA	Mandatory. The BG where the parking lot is assigned. Length is 30 chars maximum.
ParkingLotName	xsd__string	NA	Mandatory. The parking lot name length is 20 chars maximum.
newParkingLotName	xsd__string	NA	Optional. A NULL value leaves the parking lot name as is. The parking lot name length is 20 chars maximum.
RecallTimer	int	180	Optional. Field that is allowed to be modified. A NULL value leaves the current value unchanged. A value in steps of 30. Minimum value: 30. Maximum value: 600
ParkSpaceStart	int	0	Optional. Field that is allowed to be modified. A NULL value leaves the current value unchanged. Value length must not exceed 5 digits. When not set starting space is at position 0
ParkSpaceEnd	int	9	Optional. Field that is allowed to be modified. A NULL value leaves the current value unchanged. Value length must not exceed 5 digits. When not set, ending space is at position 9

Table 547 Update Parking Lot Request Fields

11.19.3 Delete Parking Lot

unsp__DeleteParkingLot deletes an existing parking lot entry inside a business group. It is placed in the Business Group menu (400) in soapCli under the general Parking Lot menu (480). SOAP performs validation of BGName and ParkingLotName. If validation is passed, SOAP destroys any association of CPS feature with parking lot. There are two type of associations:

Subscribers with CPS feature assigned indirectly with feature profiles:

- Before deleting a parking lot, SOAP will use SDAL API getLotFeatureProfileList() to query all feature profiles that use this parking lot.
- For all feature profiles associated with this parking lot, a call to the UpdateFeatureProfile2() function will take place and the CPS feature is removed. The CPS feature is removed from the feature profiles using RSS API remProfile(). Any association of feature profiles and the parking lot marked for deletion is then destroyed.
- SOAP calls the existing SDAL API removeSvcEntryFromDbAndCache() to remove the CPS feature from each subscriber that has the CPS feature propagated through feature profile. Any association of business group subscribers (that used the CPS feature through feature profiles) and the parking lot marked for deletion is then destroyed.

Subscribers with CPS feature assigned directly:

- SDAL API getLotSubscriberList() is used to query all BGL subscribers that have the CPS feature assigned with this parking lot.
- SOAP calls the existing SDAL API removeSvcEntryFromDbAndCache() to remove the CPS feature from each BGL subscriber previously found. Any association of business group subscribers and the parking lot marked for deletion is then destroyed.

Table 548 lists the return codes for this request.

Unsp__DeleteParkingLotResult		
Attribute Name	**Values**	**Messages**
Unsp__ResultCodeStruct	Success	Parking Lot <name> deleted Successfully
	Failure	Parking Lot was not deleted • Invalid BG name • Invalid Parking Lot name (syntax validation failed) • Parking Lot does not exist • BG does not exist

Table 548 Delete Parking Lot Result Fields

Table 549 lists the input fields for this request.

Unsp__DeleteParkingLotRequest			
Attribute Name	**Type**	**Default Value**	**Description**
BGName	unsp__BGName	NA	Mandatory. The BG where the parking lot is assigned. Length is 30 chars maximum.
ParkingLotName	xsd__string	NA	Mandatory. The parking lot name length is 20 chars maximum.

Table 549 Delete Parking Lot Request Fields

11.19.4 Query Parking Lot Information

Unsp__QueryParkingLotInfo queries the details of parking lot entity inside a business group. Unsp__QueryParkingLotInfo is placed in the Business Group menu (400) in soapCli under the general ParkingLot menu (480).

When calling this method, SDAL API getLotFeatureProfileList() is used to query all feature profiles that use this parking lot. The size of the returned list is saved to the AssociatedFPListSize.

SDAL API getLotSubscriberList() is then used to query all BGL subscribers that have the CPS feature assigned with this parking lot. The size of the returned list is saved to the AssociatedBGLListSize. SDAL API getLotDataList() is used for the query with the parking lot name and the BG index as inputs, and the desired parking lot structure information is returned at the output.

Below a table for the required/optional elements of the request and a table for the result is presented.

Table 550 lists the return codes for this request.

Unsp__QueryParkingLotInfoResult		
Attribute Name	**Values**	**Messages**
Unsp__ResultCodeStruct	Success	
	Failure	1. BG is found: – Invalid BG name 2. BG is found Lot is not found – Parking Lot was not found
Unsp__ParkingLotEntryResult		
Unsp__ParkingLotEntry		• BGName • ParkingLotName
Unsp__ParkingLotEntryData		• RecallTimer • ParkSpaceStart • ParkSpaceEnd • AssociatedFPListSize : Number of Feature Profiles using current Lot (see note) • AssociatedBGLListSize: Number of business group subscribers using current Lot (see note)
Unsp__ResultCodeStruct	Success, Failure	

Table 550 Query Parking Lot Info Result Fields

Table 551 lists the input fields for this request.

Unsp__QueryParkingLotInfoRequest			
Attribute Name	**Type**	**Default Value**	**Description**
BGName	unsp__BGName	NA	Mandatory. The BG where the Parking OT is assigned. Length is 30 chars maximum.
ParkingLotName	xsd__string	NA	Mandatory. The parking lot name length is 20 chars maximum.

Table 551 Query Parking Lot Info Request Fields

11.19.5 Query Parking Lot List

Unsp__QueryParkingLotList queries the list of parking lot entries. It is placed in the Business Group menu (400) in soapCli under the general ParkingLot menu (480).

Wildcards are utilized in order to provide a more versatile way for queries. Paging mechanism is supported in order to display multiple entries appropriately. SDAL API getLotDataList() is used for list query with arguments of the following table as input.

The design will supports querying by using only BG name and parking lot name.

Table 552 lists the return codes for this request.

Unsp__QueryParkingLotListResult		
Attribute Name	**Values**	**Messages**
Unsp__ResultCodeStruct	Success	Parking Lot <name> deleted Successfully
	Failure	• No Parking Lot was found
Unsp__ParkingLotEntriesResult		
LotEntry	ParkingLotName	• Invalid BG name • Invalid Parking Lot name
Unsp__ResultCodeStruct	Success, Failure	

Table 552 Query Parking Lot Info and Parking Lot Entries Result Fields

Table 553 lists the input fields for this request.

Unsp__QueryParkingLotListRequest			
Attribute Name	**Type**	**Default Value**	**Description**
BGName	unsp__BGName	NA	Mandatory. The BG where the Parking OT is assigned. Length is 30 chars maximum.
ParkingLotName	xsd__string	NA	Optional. The parking lot name length is 20 chars maximum. NIL selects all parking lots. (passes NULL to SDAL). Supports Wildcards "*" and "?" at the end of the word. The parking lot name length is 20 chars maximum.

Table 553 Query Parking Lot Info Request Fields

Note: List method displays the names of only parking lots contained in the business group. Use the QueryParkingLotInfo for additional details on each parking lot.

11.19.6 Query Parking Lot Usage List

QueryLotUsageList API provides the clients with the list of Lots. There shall be provided two different criteria for filtering the list of parking lots.

Filter by Subscriber: All Lots allocated for Subscribers are displayed.

- Filter by providing Endpoint Profile: All Lots allocated for Endpoint Profiles are displayed.

The API will support sorting by either Subscriber Name or Endpoint Profile. Also paging will be supported.

When subscriber is used as a filter, the result is a list of entries where each entry consists of the subscriber name and its internal display name.

Table 554 lists the return codes for this request.

WSDL Version	Obsolete in WSDL Vers.	Return Codes		Messages
V4.00.04.01		subRC_SUCCESS	0	
		subRC_APP_BG_DOES_NOT_EXIST	30301000	BG does not exist
		subRC_APP_BG_NAME_LEN_INVALID	30301001	Invalid BG name length
		subRC_DB_BG_LOOKUP_FAILED	30301002	BG name look up failed
		subRC_APP_BG_NAME_INVALID	30301003	Invalid BG name:
		subRC_APP_PARKING_LOT_NAME_INVALID	33000000	Invalid Parking Lot name:
		subRC_APP_PARKING_LOT_NAME_ DOES_NOT_EXIST	33000001	Parking Lot does not exist
		subRC_APP_PARKING_LOT_NAME_ ALREADY_EXISTS	33000002	Parking Lot already exists
		subRC_APP_PARKING_LOT_NAME_INVALID_L ENGTH	33000003	Invalid Parking Lot name length:
		subRC_APP_RECALL_TIMER_INVALID	33000004	Invalid Recall timer Value. Allowed values 30-600 with stepping of 30
		subRC_APP_PARKING_SPACE_START_ INVALID	33000005	Invalid ParkSpaceStart value. Value length must not exceed 5 digits
		subRC_APP_ PARKING_SPACE_END_INVALID	33000006	Invalid ParkSpaceEnd value. Value length must not exceed 5 digits
		subRC_APP_PARKING_SPACES_EXCEED_ THRESHOLD	33000007	Fail to create Parking Lot. Limit of 10000 Parking Spaces per system is exceeded.
		subRC_APP_NEW_PARKING_LOT_NAME_ ALREADY_EXISTS	33000008	Proposed name for ParkingLotName already exists. Please use a different ParkingLotName or delete the existing one.
		subRC_APP_SUB_NOT_BELONG_BG	30100006	Attempt to assign CPS feature, but subscriber not part of BG

Table 554 Parking Lot Request Return Codes

WSDL Version	Obsolete in WSDL Vers.	Return Codes		Messages
		subRC_APP_FEAT_CPS_DEP_MISSING	33000015	CPS requires BgCT when Parking Allowed is set true
		subRC_APP_FEAT_CPS_INVALID_SETTINGS	31158001	Setting both Parking Allowed and Retrieval Allowed to false is forbidden
		subRC_APP_FEAT_CPS_INVALID_PARKING_LOT_NAME	33000016	Parking Lot Name is Invalid
		subRC_APP_FEAT_CPS_MISSING_PARKING_LOT_NAME	33000017	Parking Lot Name is Required

Table 554 Parking Lot Request Return Codes

Table 555 lists the input fields for this request.

Unsp__QueryParkingLotListRequest			
Attribute Name	**Type**	**Default Value**	**Description**
OperatorId	I/F operator	yes	Char array
ParkingLotName	Parking Lot Name	yes	xsd__string
BGName	Business Group Name	yes	Xsd__string
UsageType	Select all Subscribers or Endpoint profiles that a lot belongs to by specifying the usage type	yes	enum
SortDirection	Provide means of sorting the results in ascending or descending order.	no	enum
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5,“Paging”	no	struct

Table 555 Query Parking Lot Usage Request Fields

11.20 Multiple Media Servers per Branch Office

Media Servers are used by OpenScape Voice to provide tones, announcements and conference channels. In order to service the subscribers within a branch, the following functionality is provided, which will assist the modifications required for setting up multiple media servers, with an option to support one or more media servers per branch controlled by the OpenScape Voice in the Data Center:

- Assign Media Server to Branch Office

- Unassign Media Server from Branch Office
- Get Media Server Assignment List

11.20.1 Assign Media Server to Branch Office

This method is used to assign a media server to a specific branch office.

Table 556 lists the return codes for this request.

WSDL Version	Return Codes
V4.00.01.30	subRC_SUCCESS
	subRC_APP_MS_MULTIPLE_BO_FOUND
	subRC_APP_MS_ENDPOINT_NOT_FOUND
	subRC_APP_MS_QUERY_ROUTES_FAILED
	subRC_APP_MS_MULTIPLE_DESTINATIONS_FOUND
	subRC_APP_MS_NO_DESTINATIONS_FOUND
	subRC_APP_MS_ENDPOINT_QUERY_ERROR
	subRC_APP_MS_DESTINATION_QUERY_ERROR
	subRC_APP_MS_DESTINATION_CREATION_ERROR
	subRC_APP_MS_ROUTE_CREATION_ERROR
	subRC_APP_MS_ROUTE_QUERY_ERROR
	subRC_APP_MS_NO_ROUTE_FOUND
	subRC_APP_MS_NO_RATE_AREA_FOUND
	subRC_APP_MS_MEDIA_SERVER_NOT_FOUND
	subRC_APP_MS_RTP_PARAM_READ_ERROR
	subRC_APP_MS_ORIG_DEST_DOES_NOT_EXIST
	subRC_APP_COMM_OPERATOR_INVALID
	subRC_APP_BG_NAME_LEN_INVALID
	subRC_APP_BO_BRANCH_OFFICE_NAME_INVALID
	subRC_APP_GW_NAME_INVALID
	subRC_APP_BO_QUERY_ERROR
	subRC_APP_BO_QUERY_ERROR
	subRC_APP_BO_DOES_NOT_EXIST
	subRC_APP_MS_MULTIPLE_BO_FOUND
	subRC_APP_BO_COULD_NOT_RETRIEVE_REPEPID
	subRC_APP_SUB_DN_DOES_NOT_EXIST
	subRC_DB_SUB_DN_LOOKUP_FAILED

Table 556 Assign Media Server to Branch Office Return Codes

Table 557 lists the input fields for this request.

Field Name	Description	Type	Default Value
Operator ID	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.	Str	Required
BranchOfficeName	The name of an existing branch office that will be unassigned	Str	Required
BGName	The name of an existing BG in which the branch office is included.	Str	Required
Media Server Name	The name of the media gateway to which the branch office will be associated.	Str	Required

Table 557 Assign Media Server to Branch Office Input Fields

11.20.2 Unassign Media Server from Branch Office

This method is used to unassign a media server from a specific branch office.

Table 558 lists the return codes for this request.

WSDL Version	Return Codes
V4.00.01.30	subRC_SUCCESS
	subRC_APP_MS_MULTIPLE_BO_FOUND
	subRC_APP_MS_ENDPOINT_NOT_FOUND
	subRC_APP_MS_QUERY_ROUTES_FAILED
	subRC_APP_MS_MULTIPLE_DESTINATIONS_FOUND
	subRC_APP_MS_NO_DESTINATIONS_FOUND
	subRC_APP_MS_ENDPOINT_QUERY_ERROR
	subRC_APP_MS_DESTINATION_QUERY_ERROR
	subRC_APP_MS_ROUTE_REMOVE_ERROR
	subRC_APP_MS_ROUTE_QUERY_ERROR
	subRC_APP_MS_NO_ROUTE_FOUND
	subRC_APP_MS_NO_RATE_AREA_FOUND
	subRC_APP_MS_MEDIA_SERVER_NOT_FOUND
	subRC_APP_MS_RTP_PARAM_READ_ERROR
	subRC_APP_MS_ORIG_DEST_DOES_NOT_EXIST
	subRC_APP_COMM_OPERATOR_INVALID
	subRC_APP_BG_NAME_LEN_INVALID
	subRC_APP_BO_BRANCH_OFFICE_NAME_INVALID
	subRC_APP_GW_NAME_INVALID
	subRC_APP_BO_QUERY_ERROR
	subRC_APP_BO_QUERY_ERROR

Table 558 Unassign Media Server from Branch Office Return Codes

WSDL Version	Return Codes
	subRC_APP_BO_DOES_NOT_EXIST
	subRC_APP_MS_MULTIPLE_BO_FOUND
	subRC_APP_BO_COULD_NOT_RETRIEVE_REPEPID
	subRC_APP_SUB_DN_DOES_NOT_EXIST
	subRC_DB_SUB_DN_LOOKUP_FAILED

Table 558 Unassign Media Server from Branch Office Return Codes

Table 559 lists the input fields for this request.

Field Name	Description	Type	Default Value
Operator ID	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.	Str	Required
BranchOfficeName	The name of an existing branch office for which the assignment will occur.	Str	Required
BGName	The name of an existing BG in which the branch office is included.	Str	Required
MediaServerName	The name of the media gateway to which the branch office will be associated.	Str	Required

Table 559 Unassign Media Server from Branch Office Input Fields

11.20.3 Get Media Server Assignment List

This method returns a list of media server associations for a specified branch office. If no filters are provided, the result is a list of unassigned media servers (used with the default Rate Area -1).

Table 560 lists the return codes for this request.

WSDL Version	Return Codes
V4.00.01.30	subRC_SUCCESS
	subRC_APP_MS_MULTIPLE_BO_FOUND
	subRC_APP_MS_ENDPOINT_NOT_FOUND
	subRC_APP_MS_QUERY_ROUTES_FAILED
	subRC_APP_MS_MULTIPLE_DESTINATIONS_FOUND
	subRC_APP_MS_NO_DESTINATIONS_FOUND
	subRC_APP_MS_ENDPOINT_QUERY_ERROR
	subRC_APP_MS_DESTINATION_QUERY_ERROR
	subRC_APP_MS_ROUTE_QUERY_ERROR
	subRC_APP_MS_NO_ROUTE_FOUND
	subRC_APP_MS_NO_RATE_AREA_FOUND

Table 560 Get Media Server Assignment List Return Codes

WSDL Version	Return Codes
	subRC_APP_MS_MEDIA_SERVER_NOT_FOUND
	subRC_APP_MS_RTP_PARAM_READ_ERROR
	subRC_APP_MS_ORIG_DEST_DOES_NOT_EXIST
	subRC_APP_COMM_OPERATOR_INVALID
	subRC_APP_BG_NAME_LEN_INVALID
	subRC_APP_BO_BRANCH_OFFICE_NAME_INVALID
	subRC_APP_GW_NAME_INVALID
	subRC_APP_BO_QUERY_ERROR
	subRC_APP_BO_QUERY_ERROR
	subRC_APP_BO_DOES_NOT_EXIST
	subRC_APP_MS_MULTIPLE_BO_FOUND
	subRC_APP_BO_COULD_NOT_RETRIEVE_REPEPID
	subRC_APP_SUB_DN_DOES_NOT_EXIST
	subRC_DB_SUB_DN_LOOKUP_FAILED

Table 560 Get Media Server Assignment List Return Codes

Table 561 lists the input fields for this request.

Field Name	Description	Type	Default Value
Operator ID	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.	Str	Required
BranchOfficeName	The name of an existing branch office for which the assignment list is required.	Str	N/A
BGName	The name of an existing BG in which the branch office is included.	Str	N/A

Table 561 Get Media Server Assignment List Output Fields

11.21 Branch Office

A branch office is an entity of related subscribers and endpoints and a sub entity of a business group (BG). Subscribers and endpoints belonging to a BG may optionally belong to a branch office. It is necessary to provide the name of the representative endpoint of the branch office to the createSubscriber or createEndpoint request in order to assign the subscriber/endpoint to the branch office. Please refer to Section 4.3, “Branch Offices”, on page 36 for more information.

Table 562 lists common return codes introduced for the branch office related methods. Other pre-existing return codes can also be returned.

WSDL Version	Obsolete in WSDL Version	Return Codes
V4.00.01.27		subRC_SUCCESS
		subRC_APP_BO_BGNAME_INVALID
		subRC_APP_BO_BRANCH_OFFICE_NAME_INVALID
		subRC_APP_BO_ENDPOINT_INVALID
		subRC_APP_BO_ENDPOINT_NOT_FOUND
		subRC_APP_BO_OFC_NOT_FOUND
		subRC_APP_BO_RA_NOT_FOUND
		subRC_APP_BO_SET_NP_FAILED
		subRC_APP_BO_NP_NOT_RESERVED
		subRC_APP_BO_USERNAME_INVALID
		subRC_APP_BO_PASSWORD_INVALID
		subRC_DB_BO_EADM_INTERNAL_ERROR
		subRC_APP_BO_CREATE_ERROR
		subRC_APP_BO_MODIFY_ERROR
		subRC_APP_BO_QUERY_ERROR
		subRC_APP_BO_REMOVE_ERROR
		subRC_APP_BO_REQUIRED_FIELD_NOT_SPECIFIED
		subRC_APP_BO_ISOPENBRANCH_NOT_SPECIFIED
		subRC_APP_BO_DIDDN_NOT_SPECIFIED

Table 562 Common Branch Office Return Codes

11.21.1 Create Branch Office

Used to create a branch office. The maximum number of branch offices supported on the system is 3000.

Table 563 lists the input fields for this request.

Field Name	Description	Length	Range	Default Value
Operator ID	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		Str	
BranchOfficeName	The name of the branch office to be created. Required	1-30	Character String	

Table 563 Create Branch Office Input Fields

Field Name	Description	Length	Range	Default Value
BGName	The name of an existing BG in which the BO will be included. Required	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		
RepresentativeEndpointName	The name of the endpoint that branch office members are to be associated with Executive Assistant group. Required	1-16	Str	
DIDPoolUpdate	Provides a means of specifying the operation to be performed on the DID pool	enum	DIDPoolUpdtDoNotUpdateDid DIDPoolUpdtAddDidDN, DIDPoolUpdtRemoveDidDN	DIDPoolUpdtDoNotUpdateDid
DidDn	HomeDn to be added as DID pool member		Str	
OfficeCode	Default office code to be used across the branch office		Str. Must be a digit sequence of maximum length 9. N.America: 6 digits Japan: 5 digits Korea: 4 or 5 digits Europe: 1 to 9 digits	
RateAreaName	Default rate area to be used across the branch office.		Str	
NumberPlanName	Default name of the numbering plan to be used across the branch office.	1..31	Character string.	
Attributes	A list of the following Boolean flags: (Attributes)			
IsOpenBranch	Indicates whether the branch office is of “OpenBranch” type	boolean	false true	false
Username	Username for the administration of OpenBranch type branch offices.	1-30	Str	
Password	Password for the administration of OpenBranch type branch offices.	8-30	Str	

Table 563 Create Branch Office Input Fields

11.21.2 Modify Branch Office

The modifyBranchOffice method is used to alter the data of a branch office belonging to a BG.

Table 564 lists the input fields for this request.

Field Name	Description	Length	Range	Default Value
Operator ID	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		Str	
BranchOfficeName	The name of the branch office to be created. Required	1-30	Character String	
NewBranchOfficeName	The new name of the Branch Office	1…30	Character String	optional
BGName	The name of an existing BG in which the BO will be included. Required	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		
DIDPoolUpdate	Provides a means of specifying the operation to be performed on the DID pool	enum	DIDPoolUpdtDoNotUpdateDid, DIDPoolUpdtAddDidDN, DIDPoolUpdtRemoveDidDN	DIDPoolUpdtDoNotUpdateDid
DidDn	HomeDn to be added as DID pool member		Str	
OfficeCode	Default office code to be used across the branch office		Str. Must be a digit sequence of maximum length 9. N.America: 6 digits Japan: 5 digits Korea: 4 or 5 digits Europe: 1 to9 digits	
RateAreaName	Default rate area to be used across the branch office.		Str	
NumberPlanName	Default name of the numbering plan to be used across the branch office.	1..31	Character string.	
Attributes	A list of the following Boolean flags: (Attributes)			
IsOpenBranch	Indicates whether the branch office is of “OpenBranch” type	boolean	false / true	false
Username	Username for the administration of OpenBranch type branch offices.	1-30	Str	
Password	Password for the administration of OpenBranch type branch offices.	8-30	Str	

Table 564 Modify Branch Office Input Fields

11.21.3 Query Branch Office

The queryBranchOfficemethod is used to retrieve a list of branch offices that match the filtering criteria passed on its interface.

Table 565 lists the output fields for this request.

Field Name	Description	Length	Range	Default Value
Operator ID	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		Str	
BranchOfficeName	The name of the branch office to be created.	1-30	Character String	
BGName	The name of an existing BG in which the BO will be included.	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		
RepresentativeEndpointName	The name of the endpoint that branch office members are to be associated with Executive Assistant group.	1-16	Str	
OfficeCode	Default office code to be used across the branch office		Str. Must be a digit sequence of maximum length 9. N.America: 6 digits Japan: 5 digits Korea: 4 or 5 digits Europe: 1 to 9 digits	
RateAreaName	Default rate area to be used across the branch office.		Str	
NumberPlanName	Default name of the numbering plan to be used across the branch office.	1..31	Character string.	
Attributes	A list of the following Boolean flags: (Attributes)			
IsOpenBranch	Indicates whether the branch office is of “OpenBranch” type	boolean	false true	
Username	Username for the Administration of OpenBranch type branch offices.	1-30	Str	
Password	Password for the Administration of OpenBranch type branch offices.	8-30	Str	
SortOrder	Defines the order by which the results are sorted (by name)	enum	SortOrderAscending, SortOrderDescending	SortOrderAscending

Table 565 Query Branch Office Output Fields

11.22 Remove Branch Office

The removeBranchOffice function is used to delete a specific branch office that belongs to the BG passed on its interface.

Table 566 lists the input fields for this request.

Field Name	Description	Length	Range	Default Value
Operator ID	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		Str	
BranchOfficeName	The name of the branch office to be created. Required	1-30	Character String	
BGName	The name of an existing BG in which the BO will be included. Required	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		

Table 566 Remove Branch Office Input Fields

11.23 Remote Call Pickup Groups

This chapter describes the Remote Call Pickup Groups. For every Local Call pickup group it shall be possible to create a list of Remote Pickup groups located in HiPath 4000 and/or HiPath 8000 private SIPQ network, i.e., a CPU Group may span across the SIPQ as long as the groups are administered within the same BG. Remote groups may belong to only one local group.

11.23.1 Add CPU Group

The AddCpuGroup API is used to create a new Call Pickup Group. The following table shows the format of AddCpuGroupRequest:

Field Name	Description	Mandatory	Type	Default Value [Range]
OperatorId	I/F operator	yes	Char array	NA [1…40]
BGName	The Name of the BG where the CPU belongs	yes	Char array	NA [1..30]
CpuGroupId	A unique Id for the Cpu Group	no	Integer	NA [1..10000]
CpuGroup	A structure containing the following data			
GroupName	A unique name for the CPU group. Unicode characters are supported.	no	Char array	NA [1..24]
NotificationDelayTimer	The notification delay timer for the CPU Group	no	Integer	0 [0..60]
CIDPresentation	This field controls whether the caller ID for all calls to the CPU group members, can be presented in the pickup notifications.	no	Enumeration	CpuGroupCIDPresentationAllowed, CpuGroupCIDPresentationRestricted

Table 567 Add CPU Group Input Fields

11.23.2 Modify CPU Group

The ModifyCpuGroup API is used to modify an existing Call Pickup Group. The following table shows the format of ModifyCpuGroupRequest:

Field Name	Description	Mandatory	Type	Default Value [Range]
OperatorId	I/F operator	yes	Char array	NA [1…40]
BGName	The Name of the BG where the CPU belongs	yes	Char array	NA [1..30]
CpuGroupId	A unique Id for the Cpu Group	yes	Integer	NA [1..10000]
CpuGroup	A structure containing the following data			
GroupName	A unique name for the CPU group. Unicode characters are supported.	no	Char array	NA [1..24]
NotificationDelayTimer	The notification delay timer for the CPU Group	no	Integer	0 [0..60]
CIDPresentation	This field controls whether the caller ID for all calls to the CPU group members, can be presented in the pickup notifications.	no	Enumeration	CpuGroupCIDPresentationAllowed, CpuGroupCIDPresentationRestricted

Table 568 Modify CPU Group Input Fields

11.23.3 Add Local CPU Members

Adds members to the local call pickup group.The Output fields are shown in the following table:

Field Name	Description	Mandatory	Type	Default Value [Range]
OperatorId	I/F operator	yes	Char array	NA [1…40]
BGName	The Name of the BG where the CPU belongs	yes	Char array	NA [1..30]
CpuGroupId	The unique Id for the Cpu Group	yes	Integer	NA [1..10000]
LocalCpuMemberDataList	A structure containing a list of LocalCpuMemberData.			
(LocalCpuMemberDataList) LocalCpuMemberData	A structure containing the following data			
(LocalCpuMemberData) ServiceID	The ServiceId of the local member	no	Char array	NA [1..30]
(LocalCpuMemberData) CIDCallsToMember	This field controls whether the caller ID for calls to the specific member, can be presented in the notifications to the other CPU group members.	no	Enumeration	CpuGroupCIDPresentationAllowed, CpuGroupCIDPresentationRestricted

Table 569 Add Local CPU Group Input Fields

11.23.4 Modify Local CPU Members

The ModifyLocalCpuMembers API is used to modify the configuration of subscribers that are members of a Call Pickup Group. The following table shows the format of ModifyLocalCpuMembersRequest:

Field Name	Description	Mandatory	Type	Default Value [Range]
OperatorId	I/F operator	yes	Char array	NA [1…40]
BGName	The Name of the BG where the CPU belongs	yes	Char array	NA [1..30]
CpuGroupId	The unique Id for the Cpu Group	yes	Integer	NA [1..10000]
LocalCpuMember DataList	A structure containing a list of LocalCpuMemberData.			
(LocalCpuMembe rDataList) LocalCpuMember Data	structure containing the following data			
(LocalCpuMembe rData) ServiceID	The ServiceId of the local member	no	Char array	NA [1..30]
(LocalCpuMembe rData) CIDCallsToMemb er	This field controls whether the caller ID for calls to the specific member, can be presented in the notifications to the other CPU group members.	no	Enumeration	CpuGroupCIDPresentatio nAllowed, CpuGroupCIDPresentatio nRestricted

Table 570 Modify Local CPU Group Input Fields

11.23.5 Remove Local CPU Members

The RemoveLocalCpuMembers API is used to remove subscribers from a Call Pickup Group The following table shows the format of RemoveLocalCpuMembersRequest:

Field Name	Description	Mandatory	Type	Default Value [Range]
OperatorId	I/F operator	yes	Char array	NA [1…40]
BGName	The Name of the BG where the CPU belongs	yes	Char array	NA [1..30]
CpuGroupId	The unique Id for the Cpu Group	yes	Integer	NA [1..10000]
LocalCpuMember DataList	A structure containing a list of LocalCpuMemberData.			
(LocalCpuMembe rDataList) LocalCpuMember Data	structure containing the following data			

Table 571 Remove Local CPU Group Input Fields

Field Name	Description	Mandatory	Type	Default Value [Range]
(LocalCpuMemberData) ServiceID	The ServiceId of the local member	no	Char array	NA [1..30]

Table 571 Remove Local CPU Group Input Fields

11.23.6 Add Remote Call Pickup Groups

The AddRemoteCpuGroups API is used to add remote cpu groups to a local cpu group. The following table shows the format of AddRemoteCpuGroupsRequest:

Field Name	Description	Mandatory	Type	Default Value [Range]
OperatorId	I/F operator	yes	Char array	NA [1…40]
BGName	The Name of the BG where the CPU belongs	yes	Char array	NA [1..30]
CpuGroupId	The unique Id for the Cpu Group	yes	Integer	NA [1..10000]
RemoteCpuGroup DataList	A structure containing a list of RemoteCpuGroupData.			
(RemoteCpuGroup DataList) RemoteCpuGroup Data	A structure containing the following data			
(RemoteCpuGroup Data) RemoteCpuGroup Number	The Number/ID of the Remote Call Pickup Group. This value must be administered as a local CPU group in a remote node.	yes	Integer	NA [1.10000]
(RemoteCpuGroup Data) RemoteCpuGroup AccessCode	The Destination Access Code of the Remote Call Pickup Group.	yes	Char array	NA [1..15]
(RemoteCpuGroup Data) RemoteCpuGroup Descriptor	A descriptor for the Remote Call Pickup Group.	no	Char array	NA [1..30]
(RemoteCpuGroup Data) RemoteCpuGroup NotificationDelayTimer	Notification Timer for Call Pick Up Objects	no	Integer	0 [0…60]
(RemoteCpuGroup Data) HipathDxGroup	Indicates whether the remote call pick up group belongs to a HiPath DX system.	no	Boolean	false/true

Table 572 Add Remote Call Pickup Groups Input Fields

11.23.7 Modify Remote Call Pickup Groups

The ModifyRemoteCpuGroups API is used to modify the remote cpu groups that have been provisioned for alocal cpu group. The following table shows the format of ModifyRemoteCpuGroupsRequest:

Field Name	Description	Mandatory	Type	Default Value [Range]
OperatorId	I/F operator	yes	Char array	NA [1…40]
BGName	The Name of the BG where the CPU belongs	yes	Char array	NA [1..30]
CpuGroupId	The unique Id for the Cpu Group	yes	Integer	NA [1..10000]
RemoteCpuGroup DataList	A structure containing a list of RemoteCpuGroupData.			
(RemoteCpuGroup DataList) RemoteCpuGroup Data	A structure containing the following data			
(RemoteCpuGroup Data) RemoteCpuGroup Number	The Number/ID of the Remote Call Pickup Group. This value must be administered as a local CPU group in a remote node.	yes	Integer	NA [1.10000]
(RemoteCpuGroup Data) RemoteCpuGroup AccessCode	The Destination Access Code of the Remote Call Pickup Group.	yes	Char array	NA [1..15]
(RemoteCpuGroup Data) RemoteCpuGroup Descriptor	A descriptor for the Remote Call Pickup Group.	no	Char array	NA [1..30]
(RemoteCpuGroup Data) RemoteCpuGroup NotificationDelayTimer	Notification Timer for Call Pick Up Objects	no	Integer	0 [0…60]
(RemoteCpuGroup Data) HipathDxGroup	Indicates whether the remote call pick up group belongs to a HiPath DX system.	no	Boolean	false/true

Table 573 Modify Remote Call Pickup Groups Input Fields

11.23.8 Remove Remote Call Pickup Groups

The RemoveRemoteCpuGroups API is used to remove remote cpu groups that have been provisioned for alocal cpu group. The following table shows the format of RemoveRemoteCpuGroupsRequest:

Field Name	Description	Mandatory	Type	Default Value [Range]
OperatorId	I/F operator	yes	Char array	NA [1…40]
BGName	The Name of the BG where the CPU belongs	yes	Char array	NA [1..30]
CpuGroupId	The unique Id for the Cpu Group	yes	Integer	NA [1..10000]
RemoteCpuGroupIdsList	A structure containing a list of RemoteCpuGroupId.			
(RemoteCpuGroupIdsList) RemoteCpuGroupId	A structure containing the following data			
(RemoteCpuGroupId) RemoteCpuGroupNumber	The Number/ID of the Remote Call Pickup Group. This value must be administered as a local CPU group in a remote node.	yes	Integer	NA [1.10000]
(RemoteCpuGroupId) RemoteCpuGroupAccessCode	The Destination Access Code of the Remote Call Pickup Group.	yes	Char array	NA [1..15]

Table 574 Remove Remote Call Pickup Groups Input Fields

11.23.9 Get Call Pickup Group List

The GetCpuGroupList API is used to retrieve the list of CPU groups that are provisioned for a Business Group. The following table shows the format of GetCpuGroupListRequest:

Field Name	Description	Mandatory	Type	Length / Range
OperatorId	I/F operator	Yes	Char array	1..40
BGName	The Business Group name.	Yes	Char array	1..30
CpuGroupId	Allows filtering by the Call Pickup group Id.	No	Integer	1.10000
CpuGroupName	Allows filtering by the Call Pickup group Name.	No	Integer	1..24
LocalMember	Allows filtering by the ServiceId of the Local Member	No	Char array	1..30
RemoteGroupNumber	Allows filtering by the Remote Group Number	No	Integer	1.10000
RemoteGroupAccessCode	Allows filtering by the Remote Group Access Code.	No	Char array	1..15
Paging	Standard paging structure			

Table 575 Get Call Pickup Group List Input Fields

Field Name	Description	Mandatory	Type	Length / Range
SortType	The sorting column.	No	Enum	SortByLcpgID/ SortByCountOfLocalMembers/ SortByCountOfRemoteMembers/ SortByTotalCount/ SortByCpuGroupName/ SortByCpuNotificationDelayTimer
SortOrder	The type of sort ordering.	No	Enum	Ascending / Descending

Table 575 Get Call Pickup Group List Input Fields

11.23.10 Get Call Pickup Group Member List

The GetCpuGroupMemberList API is used to retrieve the list of CPU groups that are provisioned for a Business Group along with the list of local members and remote Call Pickup Groups. The following table shows the format of GetCpuGroupMemberListRequest:

Field Name	Description	Mandatory	Type	Length / Range
OperatorId	I/F operator	Yes	Char array	1..40
BGName	The Business Group name.	Yes	Char array	1..30
CpuGroupId	Allows filtering by the Call Pickup group Id.	No	Integer	1.10000
CpuGroupName	Allows filtering by the ServiceId of the Local Member	No	Integer	1..24
LocalMember	Allows filtering by the Remote Group Number	No	Char array	1..30
RemoteGroupNumber	Allows filtering by the ServiceId of the Local Member	No	Integer	1.10000
RemoteGroupAccessCode	Allows filtering by the Remote Group Access Code.	No	Char array	1..15
Paging	Standard paging structure			
SortType	The sorting column.	No	Enum	SortByLcpgID/ SortByCountOfLocalMembers/ SortByCountOfRemoteMembers/ SortByTotalCount/ SortByCpuGroupName/ SortByCpuNotificationDelayTimer
SortOrder	The type of sort ordering.	No	Enum	Ascending / Descending

Table 576 Get Call Pickup Group Member Input Fields

11.23.11 Get Local CPU Members List

The GetLocalGroupMembersList API is used to retrieve the list of the local members of a Call Pickup Group. The following table shows the format of GetLocalCpuMemberList:

Field Name	Description	Mandatory	Type	Length / Range
OperatorId	I/F operator	Yes	Char array	1..40
BGName	The Business Group name.	Yes	Char array	1..30
CpuGroupId	The Call Pickup group Id.	Yes	Integer	1.10000
Paging	Standard paging structure			
SortOrder	The type of sort ordering.	No	Enum	Ascending / Descending
SortType	The sorting column.	No	Enum	SortByLcpgID/ SortByCountOfLocalMembers/ SortByCountOfRemoteMembers/ SortByTotalCount/ SortByCpuGroupName/ SortByCpuNotificationDelayTimer
SortOrder	The type of sort ordering.	No	Enum	Ascending / Descending

Table 577 Get Local Call Pickup Group Member Input Fields

11.23.12 Get Remote CPU Members List

The GetRemoteCpuGroupList API is used to retrieve the list of the remote groups of a Call Pickup Group. The following table shows the format of GetRemoteCpuGroupList:

Field Name	Description	Mandatory	Type	Length / Range
OperatorId	I/F operator	Yes	Char array	1..40
BGName	The Business Group name.	Yes	Char array	1..30
CpuGroupId	The Call Pickup group Id.	Yes	Integer	1.10000
RemoteCpuGroup Number	The Number/ID of the Remote Call Pickup Group. This value must be administered as a local CPU group in a remote node.	Yes	Integer	1.10000
RemoteCpuGroup AccessCode	The Destination Access Code of the Remote Call Pickup Group.	Yes	Char array	1..15
Paging	Standard paging structure			
SortOrder	The type of sort ordering.	No	Enum	Ascending / Descending

Table 578 Get Remote Call Pickup Group Member Input Fields

Field Name	Description	Mandatory	Type	Length / Range
SortType	The sorting column.	No	Enum	SortByGroupNumber/ SortByGroupAccessCode/ SortByHiPathDXGroup

Table 578 Get Remote Call Pickup Group Member Input Fields

11.24 Community Groups

Community Groups are used to configure bilateral Intercom Calling capabilities and to provide fast dialing for Intercom Calls. Any OSV subscriber that shall originate or terminate Intercom calls, has to be provisioned with a personal Community Group, defining the allowed Intercom targets/origins. Each OSV subscriber can be provisioned with no more that one community group.Table 562 lists common return codes introduced for the branch office related methods. Other pre-existing return codes can also be returned.

Table 579 lists common return codes introduced for the community group related methods. Other pre-existing return codes can also be returned.

WSDL Version	Obsolete in WSDL Version	Return Codes
V5.00.16.01		subRC_SUCCESS subRC_APP_COM_GROUP_INVALID_PERIODIC_TONE subRC_APP_COM_GROUP_SDAL_ERROR subRC_APP_COM_GROUP_ALREADY_EXISTS subRC_APP_COM_GROUP_CREATION_FAILED subRC_APP_COM_GROUP_NOT_FOUND subRC_APP_COM_GROUP_MODIFICATION_FAILED subRC_APP_COM_GROUP_REMOVAL_FAILED subRC_APP_COM_GROUP_MEMBER_INVALID_COM_NR subRC_APP_COM_GROUP_MEMBER_INVALID_DEST_NR subRC_APP_COM_GROUP_MEMBER_INVALID_DEST_NAME subRC_APP_COM_GROUP_MEMBER_SDAL_ERROR subRC_APP_COM_GROUP_MEMBER_CREATION_FAILED subRC_APP_COM_GROUP_MEMBER_COM_NR_IN_USE subRC_APP_COM_GROUP_MEMBER_NOT_FOUND

Table 579 Common Community Groups Return Codes

11.24.1 Add Community Group

The AddCommunityGroup function is used to create a new Community Group and assign it to a subscriber. A subscriber may have only one community group assigned.

Table 580 lists the input fields for this request.

Field Name	Description	Required	Length	Range	Default Value
OperatorId	I/F operator	Yes	Char array	40 (1…40)	N/A
SubscriberDN	The subscriber that the community group will be assigned to. Please refer to field Service Id of Table 10 on page 80.	Yes			
PeriodicTone	The interval of the periodic tone in seconds used to indicate that an intercom call is active. The periodic tone applies to all terminating two-way Intercom Calls.	No	Integer	0=none 5-180	120

Table 580 Add Community Group Input Fields

11.24.2 Modify Community Group

The ModifyCommunityGroup function is used to modify an existing Community Group.

Field Name	Description	Required	Length	Range	Default Value
OperatorId	I/F operator	Yes	Char array	40 (1…40)	N/A
SubscriberDN	The subscriber that the community group is assigned to. Please refer to field Service Id of Table 10 on page 80.	Yes			
PeriodicTone	The interval of the periodic tone in seconds used to indicate that an intercom call is active.	No	Integer	0=none 5-180	120

Table 581 Modify Community Group Input Fields

11.24.3 Remove Community Group

The RemoveCommunityGroup function is used to delete an existing Community Group.

Field Name	Description	Required	Length	Range	Default Value
OperatorId	I/F operator	Yes	Char array	40 (1…40)	N/A
SubscriberDN	The subscriber that the community group is assigned to. Please refer to field Service Id of Table 10 on page 80.	Yes			

Table 582 Remove Community Group Input Fields

11.24.4 Query Community Group

The QueryCommunityGroup function is used to retrieve the information about the community groups that belong to a business group. The function requires that a business group is provided in order to query for its community groups.

Field Name	Description	Required	Length	Range	Default Value
OperatorId	I/F operator	Yes	Char array	1…40	N/A
BgName	The Business Group that the Community Group belongs. Please refer to Section 5.4, “Primitive Data Definitions”, on page 80 for a detailed description.	Yes			
SubscriberDN	The subscriber that the community group is assigned to. Used for filtering for a specific community group. Please refer to field Service Id of Table 10 on page 80.	No			
SortOrder	Defines the order by which the results are sorted (by name)	No	enum	SortOrderAscending, SortOrderDescending	SortOrderAscending
Paging	Standard paging structure for Paged queries.				N/A

Table 583 Query Community Group Input Fields

Field Name	Description	Required	Length	Range	Default Value
PaginatedInfo	Standard paging structure. For details, refer to Section 5.5, “Paging”, on page 88.	Char array	1…40	N/A	PaginatedInfo
ComGroupList	An array of ComGroupElement items, each of which contains the following fields:				ComGroupList
ComGroupElement.SubscriberDN	The subscriber that the community group is assigned to. Please refer to field Service Id of Table 10 on page 80.				ComGroupElement .SubscriberDN
ComGroupElement.PeriodicTone	The interval of the periodic tone in seconds used to indicate that an intercom call is active.	Integer	0=none 5-180	120	ComGroupElement .PeriodicTone

Table 584 Query Community Group Output Fields

11.24.5 Add Community Group Member

The AddCommunityGroupMember function is used to add members to a Community Group

Field Name	Description	Required	Length	Range	Default Value
OperatorId	I/F operator	Yes	Char array	40 (1…40)	N/A
SubscriberDN	The subscriber that the community group is assigned to. Please refer to field Service Id of Table 10 on page 80.	Yes			
COMNumber	The COM# of the community group member is the (index) number to be dialed after the one-way or two-way intercom access code.	Yes	Integer	0..99	0
DestinationNumber	The destination number corresponding to the member of the community group. The destination number must refer to a translatable/routable H8K destination.	Yes	Char array	30 (0..30)	N/A
DestinationName	A description to the destination number.	No	Char array	50 (0..50)	N/A

Table 585 Add Community Group Member Input Fields

Field Name	Description	Required	Length	Range	Default Value
IncomingBlock	Indicates whether an intercom call from this number is allowed or blocked.	No	Boolean	False, True	false

Table 585 Add Community Group Member Input Fields

11.24.6 Modify Community Group Member

The ModifyCommunityGroupMember function is used to modify members in a Community Group

Field Name	Description	Required	Length	Range	Default Value
OperatorId	I/F operator	Yes	Char array	40 (1…40)	N/A
SubscriberDN	The subscriber that the community group is assigned to. Please refer to field Service Id of Table 10 on page 80.	Yes			
COMNumber	The COM# of the member of the community group that is modified.	Yes	Integer	0..99	0
NewCOMNumber	The new COM# that will be assigned to the community group member.	Yes	Integer	0..99	0
DestinationNumber	The destination number corresponding to the member of the community group. The destination number must refer to a translatable/routable H8K destination.	Yes	Char array	30 (0..30)	N/A
DestinationName	A description to the destination number.	No	Char array	50 (0..50)	N/A
IncomingBlock	Indicates whether an intercom call from this number is allowed or blocked.	No	Boolean	False, True	false

Table 586 Modify Community Group Member Input Fields

11.24.7 Remove Community Group Member

The RemoveCommunityGroupMember function is used to remove members from a Community Group

Field Name	Description	Required	Length	Range	Default Value
OperatorId	I/F operator	Yes	Char array	40 (1…40)	N/A
SubscriberDN	The subscriber that the community group is assigned to. Please refer to field Service Id of Table 10 on page 80.	Yes			
COMNumber	The COM# of the member of the community group to be deleted.	Yes	Integer	0..99	0

Table 587 Remove Community Group Member Input Fields

11.24.8 Query Community Group Member

The QueryCommunityGroupMember function is used to retrieve the information about the members of a community group.

Field Name	Description	Required	Length	Range	Default Value
OperatorId	I/F operator	Yes	Char array	1…40	N/A
SubscriberDN	The subscriber that the community group is assigned to. Used for filtering for a specific community group. Please refer to field Service Id of Table 10 on page 80.	Yes			
DestinationNumber	The destination number corresponding to the member of the community group. Allows filtering for querying a specific member.	No	Char array	30 (0..30)	N/A
SortOrder	Defines the order by which the results are sorted (by name)	No	enum	SortOrderAscending, SortOrderDescending	SortOrderAscending
SortType	Defines whether the returned members will be sorted by their COMNumber or their Destination Number.	No	enum	COM_SortBy_ComNumber, COM_SortBy_DestNumber	COM_SortBy_ComNumber
Paging	Standard paging structure for Paged queries.				N/A

Table 588 Query Community Group Member Input Fields

Field Name	Description	Required	Length	Range	Default Value
PaginatedInfo	Standard paging structure. For details, refer to Section 5.5, "Paging", on page 88.	Char array	1…40	N/A	PaginatedInfo
ComGroupMemberList	An array of ComGroupMemberElement items, each of which contains the following fields:				ComGroupMember List
ComGroupMemberList. COMNumber	The COM# of the community group member.	Integer	0..99	N/A	ComGroupMember List. COMNumber
ComGroupMemberList. DestinationNumber	The destination number of the community group member.	Char array	30 (0..30)	N/A	ComGroupMember List. DestinationNumber
ComGroupMemberList. DestinationName	The description of the destination number.	Char array	50 (0..50)	N/A	ComGroupMember List. DestinationName
ComGroupMemberList. IncomingBlock	Indicates whether an intercom call from this member is allowed or blocked	Boolean	False, True	N/A	ComGroupMember List. IncomingBlock

Table 589 Query Community Group Member Output Fields

11.25 BG Suite

This feature will allow configuring Business Group (BG) in Small Office Home Office (SOHO) with an objective of easy creation, very simple Private Number Plan (PNP) etc.

Creation of BG in SOHO will be in one or two steps with all the features unlike Business Connect (BC) solution that is used to create large and sophisticated Business Groups. Configuration of BG Services and BG lines will be same as before. However, during the BG creation, DNs can be reserved to be used by a BG so that no others can use these DNs and avoid overlapped extension dialing. While creating a BGL only one of the reserved DNs can be requested to be associated with the BGL.

The BG Suite concept composes of several existing Business Group related objects, and the new DN Reservation object, and groups them into in one single request. This aims to simply the Business Group Administrator task.

Reservation of HomeDNs and creation of routing entries at the time the Business Group is created is provided as an option. For this purpose, the CreateBGSuite supports creation of a BG and respective routing entries even when the DNReserveEnabled is set to false. All HomeDNs defined in this field must be unreserved and Vacant, else the operation shall fail.

During un-reservation of a HomeDN range, a validation of the range that is already reserved for the Business Group is performed. If the HomeDN range is reserved for a different Business Group than the one provided in the request the operation fails.

It should be noted that although private numbering plan field is optional when omitted, a new private numbering plan is created with the following convention name: NP_<<BGNAME>> with all additional provisioning of prefix access codes, destination codes and extensions. In case the creation of private numbering plan fails, BG Suite creation is aborted.

11.25.1 Create BG Suite

WSDL Version	Obsolete in WSDL Version	Return Codes
12.00.01.021		subRC_SUCCESS subRC_APP_BG_BGINFO_DOES_NOT_EXIST
V5.00.01.10		subRC_APP_BG_OVERLAP_INVALID

Table 590 Return Code List for Create BG Suite

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 80			
BGName	Please refer to Section 5.4, “Primitive Data Definitions”, on page 80			
NumberPlan Name	Private Number Plan. Refer to Section 5.4, “Primitive Data Definitions”, on page 80.	no	Char array	System Default NP PSTN – E164NANP – PNP DEFAULT [1-31]
Display Number	When a subscriber with only a private number makes a external BG call, the BG’s DisplayNumber is presented as the calling number.	yes	numeric	NA
BGAccessCode	Indicates the BG Access Code for the specifig BG.	no	Numeric string (0..9), plus * and #.	NA
OfficeCode	The digit sequence at the beginning of a range of directory numbers that represents the office in which these DNs are defined.	no	Numeric string (0..9)	NA

Table 591 Create BG Suite Input Fields

Field Name	Description	Length	Range	Default Value
StartCode / Directory Number Start	Starting Home Dn	no	Numeric string (0..9)	NA
EndCode/ Directory Number End	Ending Home Dn	no	Numeric string (0..9)	NA
Extension Length	The extension that the user will dial	no	Numeric string (0..9)	[Start/End Home DN Width …12] Default : Equal to the width of Start/End Home Dn Code
Extention Prefix Code	Prefix access codes to facilitate extension dialing	no	Numeric string (0..9)	[0…15]
Public Network Access Code			Numeric string (0..9)	9
DNReserveEnabled	Reserve DN Numbers. (WSDL V5.00.01.10)	no	boolean	false

Table 591 Create BG Suite Input Fields

11.25.2 Configuration of the Extension Dialing

During the Quick Add Business Group Task the administrator has the option to provide the Public Network Access Code and the Extension Length under the Routing Entries section. If the administrator doesn't provide input for the Extension Length, this will be interpreted by the SOAP server as setting an Extension Length value such that when added to the Office Code Length, the result is equal to the length of the full DN (Overlap=0). CreateBGSuite SOAP command configures the extension dialling by creating Prefix Access Code(s), Destination Code(s) and Extensions.

11.25.3 Prefix Access Codes

A Prefix Access Code is created for the digit located in the Extension Length position within the HomeDn directory number starting counting from the end. The digits of starting Home Dn and ending Home Dn are extracted and compared. If they match, only one Prefix Access Code is created. If they are not the same both values and their intermediaries are created.

PAC configuration for all cases above is the following:

Prefix Access Code	min/max length	Digit position	Prefix Type	Nature Of Address
Digit extracted from DN at digit position: 'Extension Length', counting from the end	Extension Length/ Extension Length	0	Extension Dialing	PNP Extension

Table 592 Prefix Access Code Configuration

11.25.4 Destination Codes for Extension dialing

For each of created Prefix Access Codes a Destination Code is created with the following configuration:

Destination Code	Nature Of Address	Destination Type
ExtensionCode (Prefix Access Code)	PNP Extension	Home Extension

Table 593 Prefix Access Code Configuration

11.25.5 Extensions

For each of the created Destination Codes an Extension is created. The Extension is calculated by finding the overlap first. The overlap is the number of digits that overlaps the extension code. It is basically the difference of Extension Length and Extension Code (starting/ending Home Dn). For the case where Extension Length is set to the minimum allowed value, that is the width of the Starting/Ending Home Dn code, (overlap =0) the first digit from Home Dn range defines the Extension Code. Additionally, if the first digit of Starting Home Dn code is not equal to the first digit of the Ending Home Dn code, multiple extensions are created by combining starting digit, ending digit and all intermediate values. For example, if Starting Dn = 2000 and Ending Dn = 5000 the produced Extensions are the 2, 3, 4 and 5. The width of the Extension is equal to the value of the overlap when the latter is positive. When overlap is zero, the Extension width is one. This means that the produced Extension Code omits always the rightmost (Extension Length -1) digits and the maximum width can be as big as the (DN total width -1).

The table below helps in understanding the rules of extension creation with an example:

Country code	Area Code	Local Office Code	Start Dn	End Dn	Extension Length	Extensions
30	210	456	1234	5678	4 (min)	1,2,3,4,5

Table 594 Extensions configuration

Country code	Area Code	Local Office Code	Start Dn	End Dn	Extension Length	Extensions
30	210	456	1234	5678	5	**6**
30	210	456	1234	5678	6	**56**
30	210	456	1234	5678	11 (max)	**0210456**

Table 594 Extensions configuration

The overlap cannot be a negative number, thus administrator is not allowed to define an extension length less than the width of the Extension Code. If this is the case, SOAP automatically defines an extension length equal to the width of the extension code, producing a zero overlap. From Assistant side, administrator is informed about the allowed range of values with an error message located at the Extension Length field.

12 Numbering Plans

A numbering plan refers to either a Private Numbering Plan (PNP) or the System Default Numbering Plan (E164NANP). The SOAP requests described in the following subsections may be used for both PNPs and for the E164NANP, where applicable.

12.1 Numbering Plan

The NumberPlanId (optional field) and NumberPlanName are the keys that logically define all the related dialing objects and tables. The Numbering Plan is the first object to be created for a PNP. All other related dialing objects, such Prefix Access Code, will be assigned with the particular NumberPlanName to be part of the PNP.

The Numbering Plan system defaults are not provisioned. A Private Numbering Plan (PNP) must be unique for each BG, except for the system default number plan.

This section describes parameters that are associated with the PNP information. The Numbering Plan specifies the format, patterns, and structure of the numbers (DN and/or Extension) used within the PNP. A PNP is used in a private network or used by a BG in a PSTN network.

12.1.1 Create Number Plan

This method creates a Number Plan.

Table 595 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.01.10		subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_PNP_NAME_MISSING subRC_APP_PNP_ID_OUT_OF_RANGE subRC_APP_PNP_NAME_LEN_INVALID subRC_APP_PNP_NAME_DUPLICATE subRC_DB_PNP_INSERTION_FAILED

Table 595 Create Number Plan Return Codes

Table 596 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
NumberPlanId	Numbering Plan Id.	integer	2 – 999 (1 is reserved for the PNP system and is the default)	N/A
NumberPlan Name	Name of the number plan. Required and cannot be changed.	1..31	Character string.	N/A
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
NumberPlan Type	Defines Numbering Plan type. For a PNP associated with a Business Group, the only value supported is NPTypePNP. Therefore, it is not necessary to be provisioned. Read-only.	enum	NPTypePNP, NPTypeE164NANP	NPTypePNP
BGName	Indicates the name of the Business Group that this Number Plan will be reserved for.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		“”

Table 596 Create Number Plan Input Fields

12.1.2 Update Number Plan

This method is used to update the BG reservation information of the Number Plan.

If the specified numbering plan is not reserved for a BG originally, it can be updated to be reserved for an existing BG. If it is already reserved for a BG, but not assigned to a BGL, it can be reserved for another existing BG or un-reserved from the currently reserved-for BG.

If the numbering plan provided is already reserved for a BG and already assigned to some BGL, it can not be reserved for another BG or un-reserved from the currently reserved-for BG.

The method can also be used for updating the Name of the Number Plan.

Table 598 lists the input fields for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.01.10		subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NUMBER_PLAN_NAME_LEN_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_NUMBER_PLAN_DOES_NOT_EXIST subRC_APP_PNP_MOD_SYSTEM_DEF_NOT_ALLOWED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_NUMBER_PLAN_DUPLICATE subRC_DB_PNP_RESERVATION_INSERTION_FAILED subRC_DB_PNP_INSERTION_FAILED subRC_DB_PNP_RESERVATION_DELETION_FAILED subRC_APP_PNP_DOES_NOT_EXIST subRC_APP_PNP_MODIFY_FAILURE subRC_APP_PNP_RSS_DB_GENERAL_FAILURE

Table 597 Update Number Plan Return Codes

Table 598 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
NumberPlan Name	Name of the number plan.	1..31	Character string.	required
NewNumberPlan Name	New Name of the number plan.	1…31	Character string	optional
BGName	This field is optional and is used to indicate the Name of the BG that this Number Plan will be reserved for If an empty string is passed the NP will be unreserved from the BG that it is reserved.	Please refer to the definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		N/A

Table 598 Update Number Plan Input Fields

12.1.3 Delete Number Plan

This method deletes a number plan with the NumberPlanName provided. Note that a number plan cannot be deleted if there are existing references to it, including if it is configured as the BG Common Number Plan for any BG.

Table 599 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.01.10		subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_PNP_NAME_MISSING subRC_APP_PNP_ID_OUT_OF_RANGE subRC_APP_PNP_NAME_LEN_INVALID subRC_APP_PNP_NAME_DUPLICATE subRC_DB_PNP_INSERTION_FAILED

Table 599 Delete Number Plan Return Codes

Table 600 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
NumberPlan Name	Name of the number plan.	1..31	Character string.	required
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			

Table 600 Delete Number Plan Input Fields

12.1.4 Get Number Plan List

This method retrieves the specified Numbering Plan information or a list of all existing Number Plans.

If a Number Plan Name is specified, only the specified Number Plan is returned if it exists. If the Number Plan Name is not specified, all the Number Plans are retrieved.

Table 601 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.01.14		subRC_SUCCESS subRC_SYS_PNP_GET_ALLOC_ERR subRC_NUMBER_PLAN_DATA_MISSING subRC_APP_PNP_NAME_LEN_INVALID subRC_APP_PNP_DOES_NOT_EXIST subRC_APP_PNP_LOOKUP_FAILED subRC_DB_PNP_LOOKUP_FAILED

Table 601 Get Number Plan List Return Codes

Table 602 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
NumberPlan Name	Name of the number plan.	1..31	Character string.	“”
InterfaceWSDL	The WSDL version of the request. This allows the SOAP Server to avoid returning fields that were introduced after the specified WSDL version.	Refer to Section 5.1.1, “The Interface WSDL and Interface Release Control”, on page 61.		N/A
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			N/A
BGName	The name of the Business Group on which to filter.	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		N/A
CombinedList	If a BG Name is specified, CombinedList indicates whether the returned list should include system wide Numbering Plans as well as those associated with the specified BG. If no BG Name is specified, CombinedList indicates whether the returned list should include all BG-related Numbering Plans as well as the system wide Numbering Plans.	Boolean	True, False	False
BGCommon	Indicates whether the list of BG Common Number Plans is to be retrieved. If the BGCommon tag is not passed, there shall be no filtering on BG Common Numbering Plan. If the BGCommon tag is passed, and the value is False, only Numbering Plans that are not BG Common Numbering Plans will be returned. If the BGCommon tag is passed, and the value is True, only Numbering Plans that are BG Common Numbering Plans will be returned.	Boolean	True, False	False

Table 602 Get Number Plan List Input Fields

12.1.5 Get Number Plan ID Info

This method retrieves the information about Numbering Plan IDs currently in use on the OpenScape Voice system. Table 603 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.01.01.14		subRC_SUCCESS subRC_INTERNAL_ERROR subRC_INVLD_FEATURE_DATA

Table 603 Get Number Plan ID Info Return Codes

Table 604 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			

Table 604 Get Number Plan ID Info Input Fields

Table 605 lists the output fields for this method.

Field Name	Description	Length	Range	Default Value
NumberOfUsedNPs	The number of Numbering Plans that have been assigned and/or reserved on the switch.	integer	N/A	N/A
NumberOfCreatedNPs	the total number of Numbering Plans that have been created on the switch.	integer	N/A	N/A
TotalNumberPlanID	The total number of Numbering Plans the switch can support.	integer	N/A	N/A
NextAvailableNumberPlanID	The lowest available Numbering Plan ID the user can use when creating the next NumberingPlan.	integer	N/A	N/A

Table 605 Get Number Plan ID Info Output Fields

12.1.6 Get Extended Numbering Plan

Currently, in order to retrieve the list of Private Numbering Plans, the following requests are made by the Assistant: GenericQuery, GetBGInfo, GenericQuery, and GetSubscriberList. From this feature on, a single new SOAP function will be introduced to handle the list of NPs, unsp__GetNumberPlanListExtended. This function will return as a minimum the following fields: NP name, Number of subscribers, whether it is default for the BG, whether it is common for the BG.

Also, we need sorting and filtering on NP Name (currently sorting is not provided) and paging of the results. RSS will provide sorting/filtering/paging facilities plus the full result-set to SOAP.

Input Parameters for Get Extended Number Plan List

Table 606 lists the output fields for this method.

Field Name	Description	Length	Range	Default Value
Operator ID	I/F Operator	Char array	40 (1…40)	
NPName	Name of the Number Plan	Char Array	31(1…31)	
BGName	Name of the Business Group on which to filter	Please refer to Section 5.4, “Primitive Data Definitions”, on page 80. N/A		
Interface WSDL	The WSDL version of the request. This allows the SOAP Server to avoid returning fields that were introduced after the specified WSDL version.			
Page Number	No	number of the retrieved result set		Integer
Page Size	No	Page size of the retrieved result set		Integer
NPSortingType	Sorting Options for Number Plan	Enum		NumberPlanStUnassigned NumberPlanStNPName,

Table 606 Get Extended Numbering Plan Input Fields

Table 607 lists the output fields for this method.

Field Name	Description	Type	Range
NPName	Name of the Number Plan	Char Array	31(1…31)
NPId	Number Plan Id stored in Database	Int	
NPType	Indication of whether NP is private or Global (E164NANP)	Enum	NPTypePNP, NPTypeE164NANP
BGName	Name of the Business Group on which to filter		
isDefault	Indication of whether the Number Plan is default for the BG	boolean	

Table 607 Get Extended Numbering Plan Input Output Fields (Sheet 1 of 2)

Field Name	Description	Type	Range
isCommon	Indication of whether the Number Plan is common for the BG	Boolean	
subCount	Number of Susbcribers associated with the Number Plan	int	

Table 607 Get Extended Numbering Plan Input Output Fields (Sheet 2 of 2)

12.2 Destination Codes

12.2.1 Create Dest Code

This method is used to create an E.164 Destination Code for the PNP. This enables the association of an E.164 Destination Code to the appropriate destination. Each E.164 Code is uniquely defined by the E.164 Code, the Numbering Plan Id, the Nature of Address, the Originating Class of Service, and the Originating Rate Area.

Note, the provisioning of the CodeIndex object is overlaid on the destination code (E164 DN code). To qualify the object being administered as a CodeIndex, a new field is introduced, CodeIndexName, which will differentiate the object from the E164 code objects.

The following parameters are applicable for administering the CodeIndex:

- Code Index Name: The symbolic name for the CodeIndex object.
- Number Plan Name: It is required to qualify the CodeIndex with the numbering plan name;
- Nature Of Address: new value, NOA_Code_Index;
- DestType: existing rules apply, plus DT_Code_Index is allowed for up to two levels of indirection CodeIndex mapping;
- The same rules apply to NPA, TrafficType, RateArea, ClassOfService, Triggers, Attributes, DestinationType, and Destination as the existing E164 codes.
- The provisioning of the E164 code object is enhanced to allow assigning the CodeIndex to the E164 code as a template for extending the routing object.
- The following parameters are applicable for provisioning an E164 code with CodeIndex:
- Digit Code: E164 digit code, 1 - 15 digits. It is required for E164 code digit and mutually exclusive of codeIndexName;

- Number Plan Name: It is required to qualify the digit code with the numbering plan name. Only a CodeIndex with the same numbering plan may be assigned to the E164 code;
- DestType: new value, DT_Code_Index, to indicate the CodeIndex mapping;
- Destination: the symbolic name of the CodeIndex to be used;
- Rate Area: allowed;
- Class Of Service: allowed;
- Triggers: allowed;
- Traffic Type: not allowed;
- Flags (attributes): not allowed.

Traffic Type Selection is applicable for administering both Destination Codes and Code Indices.

Table 608 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.01.14		subRC_APP_COMM_OPERATOR_INVALID subRC_APP_PNP_DEST_CODE_MISSING subRC_APP_PNP_DEST_CODE_LEN_INVALID subRC_APP_PNP_NAME_MISSING subRC_APP_PNP_NAME_LEN_INVALID subRC_APP_PNP_DEST_CODE_NATURE_ADDR_MISSING subRC_APP_PNP_DEST_CODE_NPA_LEN_INVALID subRC_APP_PNP_DEST_CODE_RATE_AREA_LEN_INVALID subRC_APP_PNP_DEST_CODE_COS_LEN_INVALID subRC_APP_PNP_DEST_CODE_NAME_LEN_INVALID subRC_APP_PNP_DEST_CODE_TYPE_MISSING subRC_APP_PNP_DEST_CODE_OFFICECODE_LEN_INVALID subRC_APP_PNP_DEST_CODE_OFFICECODE_LEN_INVALID subRC_APP_PNP_DEST_CODE_NEWCODE_LEN_INVALID subRC_APP_PNP_DEST_CODE_NEWCODE_CODEINDEX_MUTUAL_EX CLUSIVE

Table 608 Create Dest Code Return Codes

Table 609 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
(DestCodeId) CodeIndexName	The symbolic name of the code index. It is required for CodeIndex and mutually exclusive of the DestCode parameter. That is, either Code index name or DestCode is allowed for administering the Destination Code, but not both.	1..31	Character string	“”
(DestCodeId) NumberPlanName	Name of the number plan.	1..31	Character string.	required
(DestCodeId) NatureOfAddress	Nature of address for this destination.	enum	NOA_None, NOA_Unknown, NOA_Subscriber, NOA_National, NOA_International, NOA_LRN_, NOA_NoDN, NOA_Test, NOA_L0, NOA_L1, NOA_L2, NOA_Extension, NOA_CallingLoc, NOA_APPServer, NOA_NRN_CPN, NOA_NRN_NSN NOA_Code_Index	NOA_None
(DestCodeId) RateArea	The rate area for this destination.	1..15	Character string	“”
(DestCodeId) COS	The class of service for this destination.	1..15	Character string	“”
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
DestCodeData	A structure containing the following attributes associated with a Destination Code.			
(DestCodeData) NPA	Numbering Plan Area (NPA) code for this destination. Note that min and max lengths for this parameter are controlled by the RTP parameters “Srx/Xdm/DnCode/npa/minLen” and “Srx/Xdm/DnCode/npa/maxLen”. The default value for both is 3.	3 (also see description)	Numeric string	“”

Table 609 Create Dest Code Input Fields (Sheet 1 of 3)

Field Name	Description	Length	Range	Default Value
(DestCodeData) TrafficType	The traffic type for this destination.	enum	TT_None, TT_International, TT_InternationalZone1, TT_Interlata, TT_Intralata, TT_TollFree, TT_DA411, TT_DAHomeNPA, TT_DAForeignNPA, TT_Emergency	TT_None
(DestCodeData) TrafficType Selection	Traffic Type selection can be either Traffic Type based or Local Toll based	Enum	BasedOnTrafficType, BasedOnLocalToll	N/A
(DestCodeData) DestType	Destination Type	enum	DT_None, DT_Destination, DT_TimeOfDay, DT_Service, DT_HomeDN, DT_EnumDestination, DT_InvalidCode, DT_Intercept, DT_HomeExtension, DT_HomeLocation, DT_CallingLocation, DT_CarrierDest, DT_PrefixDest DT_Code_Index, DT_NewCode DT_CodeProcessing	required
(DestCodeData) DestName	The name of the destination.	1..15	Character string	“”
(DestCodeData) HomeDNOfficeCode	The office code of the Home DN for this destination. Note that min and max lengths for this parameter are controlled by the RTP parameters “Srx/Xdm/HomeDn/digits/minLen” and “Srx/Xdm/HomeDn/digits/maxLen”. The default values are 4 and 5 respectively.	4..5 (also see description)	Numeric string	“”
(DestCodeData) Flags	A structure containing the following list of Boolean attributes to provide further data about the destination code. Applicable only for system default numbering plan (E164NANP).			
(DestCodeData.Flags) OperPrefixNotAllowed	Indicates whether an operator prefix is allowed for the specified destination code.	boolean	true, false	false
(DestCodeData.Flags) NationalPrefixNotAllowed	Indicates whether a national prefix is allowed for the specified destination code.	boolean	true, false	false

Table 609 Create Dest Code Input Fields (Sheet 2 of 3)

Field Name	Description	Length	Range	Default Value
(DestCodeData.Flags) NationalPrefixRequired	Indicates whether a national prefix is required for the specified destination code.	boolean	true, false	false
(DestCodeData.Flags) CACNotAllowed	Indicates whether a Customer Access Code (CAC) is allowed for the specified destination code.	boolean	true, false	false
(DestCodeData.Flags) INTrigProcRequired	Indicates whether a IN trigger processing is required for the specified destination code.	boolean	true, false	false
(DestCodeData) NewCode	A DN to be used in support of the “new code” translation and routing capability. Calls to this destination are retranslated using the number in this field based on the calling party’s rate area. Only allowed (and required) if DestType is NewCode. May contain the characters 0-9, A-E, * and #.	1..30	Character String	“”
(DestCodeData) CountryCode	A country code for the destination.	1..4	Character String	“”

Table 609 Create Dest Code Input Fields (Sheet 3 of 3)

12.2.2 Modify Dest Code Info

This method is used to change the Destination Code information. Only the NPA, the Destination Name, Destination Type, the TrafficType, and/or the Home DN Office Code can be updated when changing the Destination Code information.

Table 610 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.01.14		subRC_APP_COMM_OPERATOR_INVALID subRC_APP_PNP_DEST_CODE_MISSING subRC_APP_PNP_DEST_CODE_LEN_INVALID subRC_APP_PNP_NAME_MISSING subRC_APP_PNP_NAME_LEN_INVALID subRC_APP_PNP_DEST_CODE_NATURE_ADDR_MISSING subRC_APP_PNP_DEST_CODE_NPA_LEN_INVALID subRC_APP_PNP_DEST_CODE_RATE_AREA_LEN_INVALID subRC_APP_PNP_DEST_CODE_COS_LEN_INVALID subRC_APP_PNP_DEST_CODE_NAME_LEN_INVALID subRC_APP_PNP_DEST_CODE_TYPE_MISSING subRC_APP_PNP_DEST_CODE_OFFICECODE_LEN_INVALID subRC_APP_PNP_DEST_CODE_OFFICECODE_LEN_INVALID subRC_INT_COMM_XDM_INTERNAL_ERROR subRC_APP_PNP_DEST_CODE_NEWCODE_LEN_INVALID subRC_APP_PNP_DEST_CODE_NEWCODE_CODEINDEX_MUTUAL_EX CLUSIVE

Table 610 Modify Dest Code Info Return Codes

Table 611 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
DestCodeId	A structure that uniquely identifies a Destination Code by the attributes below.			
(DestCodeId) DestCode	The numeric code of the destination. Note that min and max lengths for this parameter are controlled by the RTP parameters “Srx/Xdm/DnCode/digits/minLen” and “Srx/Xdm/DnCode/digits/maxLen”. The default values are 1 and 15 respectively.	1..15 (also see description)	Numeric string	required
(DestCodeId) CodeIndexName	For Code Index entries, the symbolic name of the code index.	1..31	Character string	“”
(DestCodeId) NumberPlanName	Name of the number plan.	1..31	Character string.	required

Table 611 Modify Dest Code Info Input Fields (Sheet 1 of 3)

Field Name	Description	Length	Range	Default Value
(DestCodeId) NatureOfAddress	Nature of address for this destination.	enum	NOA_None, NOA_Unknown, NOA_Subscriber, NOA_National, NOA_International, NOA_LRN_, NOA_NoDN, NOA_Test, NOA_L0, NOA_L1, NOA_L2, NOA_Extension, NOA_CallingLoc NOA_APPServer, NOA_NRN_CPN, NOA_NRN_NSN, NOA_Code_Index	required
(DestCodeId) RateArea	The rate area for this destination.	1..15	Character string	“”
(DestCodeId) COS	The class of service for this destination.	1..15	Character string	“”
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
DestCodeData	A structure containing the following attributes associated with a Destination Code.			
(DestCodeData) NPA	Numbering Plan Area (NPA) code for this destination. Note that min and max lengths for this parameter are controlled by the RTP parameters “Srx/Xdm/DnCode/npa/minLen” and “Srx/Xdm/DnCode/npa/maxLen”. The default value for both is 3.	3 (also see description)	Numeric string	“”
(DestCodeData) TrafficType	The traffic type for this destination.	enum	TT_None, TT_International, TT_InternationalZone1, TT_Interlata, TT_Intralata, TT_TollFree, TT_DA411, TT_DAHomeNPA, TT_DAForeignNPA, TT_Emergency	TT_None
TrafficType Selection	Traffic Type selection can be either Traffic Type based or Local Toll based	Enum	BasedOnTrafficType, BasedOnLocalToll	N/A

Table 611 Modify Dest Code Info Input Fields (Sheet 2 of 3)

Field Name	Description	Length	Range	Default Value
(DestCodeData) DestType	Destination Type	enum	DT_None, DT_Destination, DT_TimeOfDay, DT_Service, DT_HomeDN, DT_EnumDestination, DT_InvalidCode, DT_Intercept, DT_HomeExtension, DT_HomeLocation, DT_CallingLocation, DT_CarrierDest, DT_PrefixDest DT_Code_Index, DT_NewCode DT_CodeProcessing	required
(DestCodeData) DestName	The name of the destination.	1..15	Character string	“”
(DestCodeData) HomeDNOfficeCode	The office code of the Home DN for this destination. Note that min and max lengths for this parameter are controlled by the RTP parameters “Srx/Xdm/HomeDn/digits/minLen” and “Srx/Xdm/HomeDn/digits/maxLen”. The default values are 4 and 5 respectively.	4..5 (also see description)	Numeric string	“”
(DestCodeData) NewCode	A DN to be used in support of the “new code” translation and routing capability. Calls to this destination are retranslated using the number in this field based on the calling party’s rate area. Only allowed (and required) if DestType is NewCode. May contain the characters 0-9, A-E, * and #.	1..30	Character String	N/A

Table 611 Modify Dest Code Info Input Fields (Sheet 3 of 3)

12.2.3 Delete Dest Code

This method deletes the specified Destination Code.

Table 612 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.01.14		subRC_APP_COMM_OPERATOR_INVALID subRC_APP_PNP_DEST_CODE_LEN_INVALID subRC_APP_PNP_NAME_MISSING subRC_APP_PNP_NAME_LEN_INVALID subRC_APP_PNP_DEST_CODE_NATURE_ADDR_MISSING subRC_APP_PNP_DEST_CODE_RATE_AREA_LEN_INVALID subRC_APP_PNP_DEST_CODE_COS_LEN_INVALID subRC_INT_COMM_XDM_INTERNAL_ERROR

Table 612 Delete Dest Code Return Codes

Table 613 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
DestCodeId	A structure that uniquely identifies a Destination Code by the attributes below.			
(DestCodeId) DestCode	The numeric code of the destination. Note that min and max lengths for this parameter are controlled by the RTP parameters “Srx/Xdm/DnCode/digits/minLen” and “Srx/Xdm/DnCode/digits/maxLen”. The default values are 1 and 15 respectively.	1..15 (also see description)	Numeric string	required
(DestCodeId) CodeIndexName	For Code Index entries, the symbolic name of the code index.	1..31	Character string	“”
(DestCodeId) NumberPlanName	Name of the number plan.	1..31	Character string.	required
(DestCodeId) NatureOfAddress	Nature of address for this destination.	enum	NOA_None, NOA_Unknown, NOA_Subscriber, NOA_National, NOA_International, NOA_LRN_, NOA_NoDN, NOA_Test, NOA_L0, NOA_L1, NOA_L2, NOA_Extension, NOA_CallingLoc NOA_APPServer, NOA_NRN_CPN, NOA_NRN_NSN, NOA_Code_Index	required
(DestCodeId) RateArea	The rate area for this destination.	1..15	Character string	“”
(DestCodeId) COS	The class of service for this destination.	1..15	Character string	“”

Table 613 Delete Dest Code Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			

Table 613 Delete Dest Code Input Fields (Sheet 2 of 2)

12.2.4 Get Dest Code List

This method is used to retrieve a specified Destination Code or a list of Destination Codes. The following filters can be specified:

- Number Plan Name
- Nature Of Address
- Originating Rate Area
- Originating Class Of Service
- Destination Name
- Destination Type

When filters are provided, all the Destination Codes that belong to the specified filter are retrieved. If a Destination Code is specified, all the Destination Codes are retrieved. Entering a filter is available with this option. If a Destination Code is specified with a wild card, all the Destination Codes that start with the specified Destination Code digits are retrieved. Entering a filter is available with this option. If neither the Destination Code nor a filter is specified, all the Destination Codes are retrieved.

If max Destination Codes is specified, all the Destination Codes are retrieved up to the maximum number of Destination Codes specified. If the max Destination Code is not specified, the maximum number of Destination Codes retrieved is 100.

Table 614 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.01.14		subRC_SUCCESS subRC_APP_PNP_DEST_CODE_LEN_INVALID subRC_APP_PNP_NAME_LEN_INVALID subRC_APP_PNP_DEST_CODE_RATE_AREA_LEN_INVALID subRC_APP_PNP_DEST_CODE_COS_LEN_INVALID subRC_APP_PNP_DEST_CODE_NAME_LEN_INVALID subRC_APP_PNP_DEST_CODE_EXIST subRC_APP_PNP_DEST_CODE_DOES_NOT_EXIST subRC_SYS_PNP_DEST_CODE_ALLOC_ERROR subRC_APP_PNP_DEST_CODE_NUM_OUT_OF_RANGE subRC_SYS_PNP_DEST_CODE_ALLOC_ERROR

Table 614 Get Dest Code List Return Codes

Table 615 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
DestCode	The numeric code of the destination to display.	1..15	Numeric string	N/A
NumberPlanName	Name of the number plan to filter on.	1..31	Character string.	N/A
NatureOfAddress	Nature of address to filter on.	enum	NOA_None, NOA_Unknown, NOA_Subscriber, NOA_National, NOA_International, NOA_LRN_, NOA_NoDN, NOA_Test, NOA_L0, NOA_L1, NOA_L2, NOA_Extension, NOA_CallingLoc, NOA_APPServer, NOA_NRN_CPN, NOA_NRN_NSN, NOA_Code_Index	N/A
RateArea	The rate area to filter on.	1..15	Character string	N/A
COS	The class of service to filter on.	1..15	Character string	N/A
MaxDestCodes	The maximum number of entries to display in the result list.	integer	1..100	N/A
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			

Table 615 Get Dest Code List Input Fields

12.2.5 Get Dest Code

This method retrieves and displays the specified Destination Code.

Table 612 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
11.00.09.01		subRC_SUCCESS

Table 616 Delete Dest Code Return Codes

Table 613 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
DestCode	The numeric code of the destination.	1..15	Numeric string	required
NumberPlanName	Name of the number plan.	1..31	Character string.	required
NatureOfAddress	Nature of address for this destination.	enum	NOA_None, NOA_Unknown, NOA_Subscriber, NOA_National, NOA_International, NOA_LRN_, NOA_NoDN, NOA_Test, NOA_L0, NOA_L1, NOA_L2, NOA_Extension, NOA_CallingLoc, NOA_APPServer, NOA_NRN_CPN, NOA_NRN_NSN, NOA_Code_Index	N/A
RateArea	The rate area for this destination.	1..15	Character string	“”
COS	The class of service for this destination.	1..15	Character string	“”
InterfaceWSDL	The WSDL version of the request. This allows the SOAP Server to avoid returning fields that were introduced after the specified WSDL version.	Refer to Section 5.1, “Interface (WSDL) Versions”, on page 61.		N/A

Table 617 Delete Dest Code Input Fields

12.2.6 Get Dest Service List

This method retrieves a list of service destinations.

Table 618 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.01.14		subRC_SUCCESS subRC_SYS_PNP_DEST_SERVICE_PAC_ALLOC_ERROR subRC_DB_PNP_DEST_SERVICE_LOOKUP_FAILED

Table 618 Get Dest Service List Return Codes

Table 619 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			N/A

Table 619 Get Dest Service List Input Field Descriptions

The output is a list of destination service entries, identified by name and type.

12.3 Prefix Access Code

Prefix Access Codes define the dialing sequence of a dialing and number plan; it is used for E.164 and PNP. This table is used to translate prefix codes, feature access codes, speed codes, other access codes, and determines the nature of address (NOA). The numberPlanId defines the prefix access codes for each numbering plan logically.

12.3.1 Create Prefix Access Code

This method creates a Prefix Access Code. The Digits, DNMinLen and DNMaxLen must identify a unique Prefix Access Code within the specified numbering plan. The remaining parameters represent additional attributes of the Prefix Access Code to be created.

Table 620 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.01.14		subRC_SUCCESS subRC_APP_PNP_PAC_DIGITS_LEN_INVALID subRC_APP_PNP_PAC_DIGITS_INVALID subRC_APP_PNP_NAME_LEN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_PNP_PAC_PREFIXTYPE_MISSING subRC_APP_PNP_PAC_NOA_MISSING subRC_APP_PNP_PAC_DEST_TYPE_PAC_DEST_INVALID subRC_APP_PNP_PAC_DEST_TYPE_PAC_SERVICE_INVALID subRC_APP_PNP_PAC_DEST_TYPE_PAC_INVALIDCODE_INVALID subRC_APP_PNP_PAC_OPERATION_INVALID subRC_APP_PNP_PAC_OPERATION_FAILED

Table 620 Create Prefix Access Code Return Codes

Table 621 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
NumberPlanName	Name of the number plan.	1..31	Character string.	required
Digits	Prefix and access code. Tuple <digits, dnMinLen, dnMaxLen> is the unique key to the table. For the ETSI market, the digits string value allows for alphabetical characters from A to E as there is a requirement for supporting Hex digits. Thus, Hex digits apply only where the market variant is ETSI.	1..15	Numeric string (0..9), plus * and #. (also see description)	required
DnMinLen	The minimum length of the digits used for digit matching.	integer	0..30	0
DnMaxLen	The maximum length of the digits used for digit matching. If the DnMinLen is 0, DnMaxLen must be 0 also. If the <digits> are the same for the different entries, the <dnMinLen, dnMaxLen> must not overlap. The <digits> with dnMinLen/dnMaxLen has priority the same <digits> without DnMinLen/DnMaxLen when checking for a match.	30	Numeric string.	0
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			

Table 621 Create Prefix Access Code Input Fields (Sheet 1 of 5)

Field Name	Description	Length	Range	Default Value
PrefixType	Indicates the type of prefix and access code. For PNP, the valid values are: • OPTR0PLUS(1) • OPTR0(2) • OPTR00(3) • INTLOPTR(4) • IDDD(5) • DDD(6) • NP(7) – Default PNP plan only. Not provisioned for customer-defined PNP. • FGD(8) • VSC(9) • SC(10) • CAC(11) – Can be an integral part of the called party number for non-North American Numbering Plan. • INVALIDCODE(17) • EXT(18) – For extension dialing, it applies to PNPs only. For system default PNP, the destId and destType do not apply. For customer-defined PNP (planId 2-999), destType is optional; only allowed value is E164DEST.	enum	PAC_Operator0Plus, PAC_Operator0, PAC_Operator00, PAC_OperatorIntl, PAC_FGD, PAC_IDDD, PAC_DDD, PAC_NoPrefix, PAC_VerticalSvcCode, PAC_SpeedCode, PAC_CarrierCode, PAC_InvalidCode_, PAC_Extension, PAC_OffNetAc, PAC_OnNetAc, PAC_FacilityAc, PAC_AttendantAc, PAC_HomeCarrierCode PAC_HomeNRN, PAC_PreventINTriggering	required

Table 621 Create Prefix Access Code Input Fields (Sheet 2 of 5)

Field Name	Description	Length	Range	Default Value
PrefixType *cont.*	• OFFNETAC(19) – public access code for PSTN outgoing calls. Applies to PNP only. For system default PNP, destId and destType do not apply. For customer-defined PNP (planId 2-999), destId and destType are optional; only allowed value is E164DEST. • ONNETAC(20) – access code for on-net calls; applies to PNPs only. The destId and destType are not applicable. • FACILITYAC(21) – private facility access code; applies to PNPs only. destType must be DEST; destId must refer to a destination in the e164_dest_t table. • ATTDNTAC(22 – attendant access code; applies to PNPs only. For the system default PNP, the destId and destType are not applicable. For customer defined PNP (planId 2-999), the destId and destType is optional and the only allowed value is E164DEST. • HomeCarrierCode (23) — Indicates that the CAC of a particular call belongs to the owner of the switch. Release 9 only.			
DigitPos	Indicates the number of leading digits to be stripped from the destination DN before translating the resulting DN. If digitPos = 0, the prefix digits are not dropped from the called number.	integer		0
CACLength	Indicates the length of the Carrier Access Code (CAC) after the digit position. In the case of a CAC prefix type, it is allowed values between 0 and 6 and has a default value of 0. This is an optional field for non-CAC Prefix Type and mandatory for CAC Prefix Type. *R9 only.*	integer		0
DigitsInsert	Inserted after the digitPos.	1..15	Numeric string	“”

Table 621 Create Prefix Access Code Input Fields (Sheet 3 of 5)

Field Name	Description	Length	Range	Default Value
TrailingDigit Pos	Indicates the position at which to delete extraneous trailing digits.	integer	Unsigned integer of range 1..30	31
noa	Indicates the Nature of Address of the DN resulting from the digit editing. The digit editing is based on the digitPos, digitsInsert, and trailingDigitPos. For PNP, the valid values are: UNKNOWN(0), SUBS(1), NATL(2), INTL(3), PRFXBASED(6), NODN(7), L0(21), L1(22), L2(23), EXT(24), PRFXBASEDNOTNS(25) • L0 – the number of PNP Level 0 (Subscriber) number. • L1 – the number of PNP Level 1 (National) number. • L2 – the number of PNP Level 2 (International) number. EXT – the number of an extension; applies to PNP only. The prefixType EXT along with dnMinLen and dnMaxLen should determine an EXT number.	enum	PAC_Unknown, PAC_Subscriber, PAC_National, PAC_International, PAC_PrefixBased, PAC_NoDn, PAC_L0, PAC_L1, PAC_L2, PAC_NoaExtension, PAC_PrefixBasedNoTNS	required

Table 621 Create Prefix Access Code Input Fields (Sheet 4 of 5)

Field Name	Description	Length	Range	Default Value
destType	Valid values are: • PAC_Dest – the call is routed to an outgoing destination. Applicable only if the prefixType is FACILITYAC(21). • PAC_Service – Applicable only if the prefixType is VSC(9) or SC(10). • PAC_InvalidCode – Applicable only if the prefixType is INVALIDCODE(17). • PAC_E164Dest – the destination is determined by translation in the E.164 plan or the system default PNP. E164DEST is applicable only to the customer-defined PNP, plan Id (2-999). The noa must be UNKNOWN(0). • PAC_BGCommonDest - directs the translation to “jump” to the BG Common Numbering Plan for further translation. Only the BG Numbering Plan is allowed to escape to the BG Common Numbering Plan, i.e. this Destination Type is not applicable to non-BG Numbering Plans • PAC_LocalToll -	enum	PAC_Undefined, PAC_Dest, PAC_Service, PAC_InvalidCode, PAC_E164Dest, PAC_BGCommonDestPAC_LocalToll	N/A

Table 621 Create Prefix Access Code Input Fields (Sheet 5 of 5)

12.3.2 Modify Prefix Access Code

This method modifies the specified prefix access code.

Table 622 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.01.14		subRC_SUCCESS subRC_APP_PNP_PAC_DIGITS_LEN_INVALID subRC_APP_PNP_PAC_DIGITS_INVALID subRC_APP_PNP_NAME_LEN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_PNP_PAC_PREFIXTYPE_MISSING subRC_APP_PNP_PAC_NOA_MISSING subRC_APP_PNP_PAC_DEST_TYPE_PAC_DEST_INVALID subRC_APP_PNP_PAC_DEST_TYPE_PAC_SERVICE_INVALID subRC_APP_PNP_PAC_DEST_TYPE_PAC_INVALIDCODE_INVALID subRC_APP_PNP_PAC_OPERATION_INVALID subRC_APP_PNP_PAC_OPERATION_FAILED

Table 622 Modify Prefix Access Code Return Codes

Table 623 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
NumberPlanName	Name of the number plan.	1..31	Character string.	required
Digits	Prefix and access code. Tuple <digits, dnMinLen, dnMaxLen> is the unique key to the table. For the ETSI market, the digits string value allows for alphabetical characters from A to E as there is a requirement for supporting Hex digits. Thus, Hex digits apply only where the market variant is ETSI.	1..15	Numeric string (0..9), plus * and #. (also see description)	required
DnMinLen	The minimum length of the digits used for digit matching.	integer	0..30	0
DnMaxLen	The maximum length of the digits used for digit matching. If the DnMinLen is 0, DnMaxLen must be 0 also. If the <digits> are the same for the different entries, the <dnMinLen, dnMaxLen> must not overlap. The <digits> with dnMinLen/dnMaxLen has priority the same <digits> without DnMinLen/DnMaxLen when checking for a match.	30	Numeric string.	0
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			

Table 623 Modify Prefix Access Code Input Fields (Sheet 1 of 5)

Field Name	Description	Length	Range	Default Value
PrefixType	Indicates the type of prefix and access code. For PNP, the valid values are: • OPTR0PLUS(1) • OPTR0(2) • OPTR00(3) • INTLOPTR(4) • IDDD(5) • DDD(6) • NP(7) – Default PNP plan only. Not provisioned for customer-defined PNP. • FGD(8) • VSC(9) • SC(10) • CAC(11) – Can be an integral part of the called party number for non-North American Numbering Plan. • INVALIDCODE(17) • EXT(18) – For extension dialing, it applies to PNPs only. For system default PNP, the destId and destType do not apply. For customer-defined PNP (planId 2-999), destType is optional; only allowed value is E164DEST.	enum	PAC_Operator0Plus, PAC_Operator0, PAC_Operator00, PAC_OperatorIntl, PAC_FGD, PAC_IDDD, PAC_DDD, PAC_NoPrefix, PAC_VerticalSvcCode, PAC_SpeedCode, PAC_CarrierCode, PAC_InvalidCode_, PAC_Extension, PAC_OffNetAc, PAC_OnNetAc, PAC_FacilityAc, PAC_AttendantAc, PAC_HomeCarrierCode, PAC_HomeNRN, PAC_PreventINTriggering	N/A

Table 623 Modify Prefix Access Code Input Fields (Sheet 2 of 5)

Field Name	Description	Length	Range	Default Value
PrefixType *cont.*	• OFFNETAC(19) – public access code for PSTN outgoing calls. Applies to PNP only. For system default PNP, destId and destType do not apply. For customer-defined PNP (planId 2-999), destId and destType are optional; only allowed value is E164DEST. • ONNETAC(20) – access code for on-net calls; applies to PNPs only. The destId and destType are not applicable. • FACILITYAC(21) – private facility access code; applies to PNPs only. destType must be DEST; destId must refer to a destination in the e164_dest_t table. • ATTDNTAC(22 – attendant access code; applies to PNPs only. For the system default PNP, the destId and destType are not applicable. For customer defined PNP (planId 2-999), the destId and destType is optional and the only allowed value is E164DEST. • HomeCarrierCode (23) — Indicates that the CAC of a particular call belongs to the owner of the switch. Release 9 only.			
DigitPos	Indicates the number of leading digits to be stripped from the destination DN before translating the resulting DN. If digitPos = 0, the prefix digits are not dropped from the called number. Optional.	integer		N/A
CACLength	Indicates the length of the Carrier Access Code (CAC) after the digit position. In the case of a CAC prefix type, it is allowed values between 0 and 6 and has a default value of 0. This is an optional field for non-CAC Prefix Type and mandatory for CAC Prefix Type. R9 only.	integer		N/A
DigitsInsert	Inserted after the digitPos.	1..15	Numeric string	N/A

Table 623 Modify Prefix Access Code Input Fields (Sheet 3 of 5)

Field Name	Description	Length	Range	Default Value
TrailingDigit Pos	Indicates the position at which to delete extraneous trailing digits. Optional.	integer	Unsigned integer of range 1.. 31	N/A
noa	Indicates the Nature of Address of the DN resulting from the digit editing. The digit editing is based on the digitPos, digitsInsert, and trailingDigitPos. For PNP, the valid values are: UNKNOWN(0), SUBS(1), NATL(2), INTL(3), PRFXBASED(6), NODN(7), L0(21), L1(22), L2(23), EXT(24), PRFXBASEDNOTNS(25) • L0 – the number of PNP Level 0 (Subscriber) number. • L1 – the number of PNP Level 1 (National) number. • L2 – the number of PNP Level 2 (International) number. EXT – the number of an extension; applies to PNP only. The prefixType EXT along with dnMinLen and dnMaxLen should determine an EXT number. Required.	enum	PAC_Unknown, PAC_Subscriber, PAC_National, PAC_International, PAC_PrefixBased, PAC_NoDn, PAC_L0, PAC_L1, PAC_L2, PAC_NoaExtension, PAC_PrefixBasedNoTNS	N/A

Table 623 Modify Prefix Access Code Input Fields (Sheet 4 of 5)

Field Name	Description	Length	Range	Default Value
destType	Valid values are: • PAC_Dest – the call is routed to an outgoing destination. Applicable only if the prefixType is FACILITYAC(21). • PAC_Service – Applicable only if the prefixType is VSC(9) or SC(10). • PAC_InvalidCode – Applicable only if the prefixType is INVALIDCODE(17). • PAC_E164Dest – the destination is determined by translation in the E.164 plan or the system default PNP. E164DEST is applicable only to the customer-defined PNP, plan Id (2-999). The noa must be UNKNOWN(0). • PAC_BGCommonDest - directs the translation to “jump” to the BG Common Numbering Plan for further translation. Only the BG Numbering Plan is allowed to escape to the BG Common Numbering Plan, i.e. this Destination Type is not applicable to non-BG Numbering Plans • PAC_LocalToll -	enum	PAC_Undefined, PAC_Dest, PAC_Service, PAC_InvalidCode, PAC_E164Dest, PAC_BGCommonDestPAC_LocalToll	N/A
destName	Indicates the name of the destination.	1..15	Character string	N/A

Table 623 Modify Prefix Access Code Input Fields (Sheet 5 of 5)

12.3.3 Delete Prefix Access Code

This method deletes the specified Prefix Access Code.

Table 624 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.01.14		subRC_SUCCESS subRC_APP_PNP_PAC_DIGITS_LEN_INVALID subRC_APP_PNP_PAC_DIGITS_INVALID subRC_APP_PNP_NAME_LEN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_PNP_PAC_DELETION_FAILED

Table 624 Delete Prefix Access Code Return Codes

Table 625 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
NumberPlanName	Name of the number plan.	1..31	Character string.	required
Digits	Prefix and access code. Tuple <digits, dnMinLen, dnMaxLen> is the unique key to the table. For the ETSI market, the digits string value allows for alphabetical characters from A to E as there is a requirement for supporting Hex digits. Thus, Hex digits apply only where the market variant is ETSI.	1..15	Numeric string (0..9), plus * and #. (also see description)	required
DnMinLen	The minimum length of the digits used for digit matching.	integer	0..30	0
DnMaxLen	The maximum length of the digits used for digit matching. If the DnMinLen is 0, DnMaxLen must be 0 also. If the <digits> are the same for the different entries, the <dnMinLen, dnMaxLen> must not overlap. The <digits> with dnMinLen/dnMaxLen has priority the same <digits> without DnMinLen/DnMaxLen when checking for a match.	30	Numeric string.	0
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			

Table 625 Delete Prefix Access Code Input Fields

12.3.4 Get Prefix Access Code List

This method retrieves all Prefix Access Codes on the system. It h as enhanced sorting capabilities based on Code and filtering by all the fields of unsp__PrefixAccessCodeEntry. The Result List accommodates the Nature of Address.

Table 626 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.01.14		subRC_SUCCESS subRC_APP_PNP_NAME_LEN_INVALID subRC_SYS_PNP_PAC_ALLOC_ERROR subRC_APP_PNP_PAC_XDM_LOOKUP_FAILED

Table 626 Get Prefix Access Code List Return Codes

Table 627 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
NumberPlanName	Name of the number plan.	1..31	Character string.	required
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			
InterfaceWSDL	Refer to Section 5.1.1, “The Interface WSDL and Interface Release Control”, on page 61.			
SortType	Sorting options criteria for Prefix Access Codes	Enum	SortByUnassigned = -1, SortByCode = 1	N/A
SortOrder	Determines the way the Objects are sorted.	Enum	SortOrderAscending - Objects aresorted in ascending order. SortOrderDescending - Objects aresorted in descending order.	SortOrderAscending
Prefix Access Code Entry Structure				
Digits	Prefix and access code. Tuple <digits, dnMinLen, dnMaxLen> is the unique key to the table. For the ETSI market, the digits string value allows for alphabetical characters from A to E as there is a requirement for supporting Hex digits. Thus, Hex Digits apply only where the market variant is ETSI.	Please refer to the definition of Digits in Section 12.3.1, “Create Prefix Access Code”, on page 483		
DnMinLen	The minimum length of the digits used for digit matching.	Please refer to the definition of DnMinLen in Section 12.3.1, “Create Prefix Access Code”, on page 483		
DnMaxLen	The maximum length of the digits used for digit matching. If the DnMinLen is 0, DnMaxLen must be 0 also. If the <digits> are the same for the different entries, the <dnMinLen, dnMaxLen> must not overlap. The <digits> with dnMinLen/dnMaxLen has priority the same <digits> without DnMinLen/ DnMaxLen when checking for a match.	Please refer to the definition of DnMaxLen in Section 12.3.1, “Create Prefix Access Code”, on page 483		
PrefixType	Indicates the type of prefix and access code.	Please refer to the definition of PrefixType in Section 12.3.1, “Create Prefix Access Code”, on		

Table 627 Get Prefix Access Code List Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
Dest Type	Indicates the type of Destination for the prefix access code	Enum	PAC_Undefined, PAC_Dest, PAC_Service, PAC_InvalidCode, PAC_E164Dest, PAC_BGCommonDest, PAC_LocalToll,	
DestName	Name of Destination	char	1…68	
Natureofaddress	Nature of Address for the Destination of Prefix Access Code	Enum	PAC_Unassigned = -1, PAC_Unknown, PAC_Subscriber, PAC_National, PAC_International, PAC_PrefixBased, PAC_NoDn, PAC_L0, PAC_L1, PAC_L2, PAC_NoaExtension, PAC_PrefixBasedNoTNS, PAC_NRN_CPN	

Table 627 Get Prefix Access Code List Input Fields (Sheet 2 of 2)

Table 628 lists the output fields for this method.

Field Name	Description	Length	Range	Default Value
PrefixAccessCodeList, a list of elements of type PrefixAccessCodeEntry, each of which contains the following attributes, uniquely identifying a prefix access code:				
Digits	Prefix and access code. Tuple <digits, dnMinLen, dnMaxLen> is the unique key to the table. For the ETSI market, the digits string value allows for alphabetical characters from A to E as there is a requirement for supporting Hex digits. Thus, Hex digits apply only where the market variant is ETSI.	Please refer to the definition of Digits in Section 12.3.1, “Create Prefix Access Code”, on page 535		NA
DnMinLen	The minimum length of the digits used for digit matching.	Please refer to the definition of DnMinLen in Section 12.3.1, “Create Prefix Access Code”, on page 535		0

Table 628 Get Prefix Access Code List Output Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
DnMaxLen	The maximum length of the digits used for digit matching. If the DnMinLen is 0, DnMaxLen must be 0 also. If the <digits> are the same for the different entries, the <dnMinLen, dnMaxLen> must not overlap. The <digits> with dnMinLen/dnMaxLen has priority the same <digits> without DnMinLen/ DnMaxLen when checking for a match.	Please refer to the definition of DnMaxLen in Section 12.3.1, “Create Prefix Access Code”, on page 535		0
PrefixType	Indicates the type of prefix and access code.	Please refer to the definition of PrefixType in Section 12.3.1, “Create Prefix Access Code”, on page 535		required
Natureofaddress	Nature of Address for the Destination of Prefix Access Code	Enum	PAC_Unassigned = -1, PAC_Unknown, PAC_Subscriber, PAC_National, PAC_International, PAC_PrefixBased, PAC_NoDn, PAC_L0, PAC_L1, PAC_L2, PAC_NoaExtension, PAC_PrefixBasedNoTNS, PAC_NRN_CPN	

Table 628 Get Prefix Access Code List Output Fields (Sheet 2 of 2)

12.3.5 Get Prefix Access Code One

This method retrieves an individual Prefix Access Code.

Table 629 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.01.14		subRC_SUCCESS subRC_APP_PNP_PAC_DIGITS_LEN_INVALID subRC_APP_PNP_PAC_DIGITS_INVALID subRC_APP_PNP_NAME_LEN_INVALID subRC_DB_PNP_PAC_LOOKUP_FAILED

Table 629 Get Prefix Access Code One Return Codes

Table 630 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
Digits	Prefix and access code. Tuple <digits, dnMinLen, dnMaxLen> is the unique key to the table. For the ETSI market, the digits string value allows for alphabetical characters from A to E as there is a requirement for supporting Hex digits. Thus, Hex digits apply only where the market variant is ETSI.	1..15	Numeric string (0..9), plus * and #. (also see description)	required
NumberPlanName	Name of the number plan.	1..31	Character string.	required
InterfaceWSDL	Refer to Section 5.1.1, “The Interface WSDL and Interface Release Control”, on page 61.			
DnMinLen	The minimum length of the digits used for digit matching. Optional.	30	Numeric string.	0
DnMaxLen	The maximum length of the digits used for digit matching. If the DnMinLen is 0, DnMaxLen must be 0 also. If the <digits> are the same for the different entries, the <dnMinLen, dnMaxLen> must not overlap. The <digits> with dnMinLen/dnMaxLen has priority the same <digits> without DnMinLen/DnMaxLen when checking for a match.	30	Numeric string.	0

Table 630 Get Prefix Access Code One Input Fields

12.4 Private Numbering Plan Location Codes

The PNP location defines the location part of the PNP. Note that these requests apply only to private numbering plans, not the public system default numbering plan.

12.4.1 Create PNP Location Code

This method creates a PNP Location Code.

Table 631 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.01.16		subRC_SUCCESS subRC_APP_PNP_LOC_CODE_LEN_INVALID subRC_APP_PNP_LOC_CODE_INVALID subRC_APP_PNP_NAME_LEN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_PNP_LOC_CODE_LOCL0LEN_INVALID subRC_APP_PNP_LOC_CODE_MODIFY_ERROR subRC_APP_PNP_LOC_CODE_OPERATION_INVALID subRC_APP_PNP_LOC_CODE_OPERATION_FAILED

Table 631 Create Pnp Location Code Return Codes

Table 632 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
locCode	Consists of L2, L1, and L0 codes. The user interfaces will have the user enter the level codes in separate digit strings that the SOAP Server will concatenate into one digit string and derive the locL0Len, locL1Len, and locL2Len. • L1 and L2 are optional. • L1 cannot be administered without L0. • L2 cannot be administered without L1.	1..14	Numeric string.	required
NumberPlanName	Name of the number plan.	1..31	Character string.	required
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
locL0Len	L0 code length.	integer	1..4	1
locL1Len	L1 code length.	integer	0..6	0
locL2Len	L2 code length.	integer	0..4	0
locSkipDigits	Indicates the number of leading digits in the location code to skip when converting the fully qualified PNP number to the extension.	integer		0

Table 632 Create Pnp Location Code Input Fields

12.4.2 Modify PNP Location Code

This method modifies the specified PNP Location Code.

Table 633 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.01.16		subRC_SUCCESS subRC_APP_PNP_LOC_CODE_LEN_INVALID subRC_APP_PNP_LOC_CODE_INVALID subRC_APP_PNP_NAME_LEN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_PNP_LOC_CODE_LOCL0LEN_INVALID subRC_APP_PNP_LOC_CODE_MODIFY_ERROR subRC_APP_PNP_LOC_CODE_OPERATION_INVALID subRC_APP_PNP_LOC_CODE_OPERATION_FAILED

Table 633 Modify Pnp Location Code Return Codes

Table 634 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
locCode	Consists of L2, L1, and L0 codes. The user interfaces will have the user enter the level codes in separate digit strings that the SOAP Server will concatenate into one digit string and derive the locL0Len, locL1Len, and locL2Len. • L1 and L2 are optional. • L1 cannot be administered without L0. • L2 cannot be administered without L1.	1..14	Numeric string.	required
NumberPlanName	Name of the number plan.	1..31	Character string.	required
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
locL0Len	L0 code length.	integer	1..4	1
locL1Len	L1 code length.	integer	0..6	0
locL2Len	L2 code length.	integer	0..4	0
locSkipDigits	Indicates the number of leading digits in the location code to skip when converting the fully qualified PNP number to the extension.	integer		0

Table 634 Modify Pnp Location Code Input Fields

12.4.3 Delete PNP Location Code

This method deletes the specified PNP Location Code.

Table 635 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.01.16		subRC_SUCCESS subRC_APP_PNP_LOC_CODE_LEN_INVALID subRC_APP_PNP_LOC_CODE_INVALID subRC_APP_PNP_NAME_LEN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_PNP_LOC_CODE_DELETION_FAILED

Table 635 Delete Pnp Location Code Return Codes

Table 636 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
locCode	Consists of L2, L1, and L0 codes. The user interfaces will have the user enter the level codes in separate digit strings that the SOAP Server will concatenate into one digit string and derive the locL0Len, locL1Len, and locL2Len. • L1 and L2 are optional. • L1 cannot be administered without L0. • L2 cannot be administered without L1.	1..14	Numeric string.	required
NumberPlanName	Name of the number plan.	1..31	Character string.	required
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			

Table 636 Delete Pnp Location Code Input Fields

12.4.4 Get PNP Location Code List

This method retrieves a list of all PNP location codes on the switch.

Table 637 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.01.16		subRC_SUCCESS subRC_SYS_PNP_LOC_CODE_ALLOC_ERROR subRC_APP_PNP_LOC_CODE_LEN_INVALID subRC_APP_PNP_LOC_CODE_INVALID subRC_APP_PNP_NAME_LEN_INVALID subRC_APP_PNP_LOC_CODE_NOT_EXIST subRC_INT_COMM_XDM_INTERNAL_ERROR subRC_DB_PNP_LOC_CODE_LOOKUP_FAILED subRC_APP_PNP_LOC_CODE_NOT_EXIST

Table 637 Get PNP Location Code List Return Codes

Table 638 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
locCode	Consists of L2, L1, and L0 codes. The user interfaces will have the user enter the level codes in separate digit strings that the SOAP Server will concatenate into one digit string and derive the locL0Len, locL1Len, and locL2Len. • L1 and L2 are optional. • L1 cannot be administered without L0. • L2 cannot be administered without L1.	1..14	Numeric string.	“”
NumberPlanName	Name of the number plan.	1..31	Character string.	required
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			N/A

Table 638 Get PnP Location Code List Input Fields

12.5 PNP Extension

The PNP Extension defines the extension part of the Private Numbering Plan. Note that these requests apply only to private numbering plans, not the public system default numbering plan.

12.5.1 Create PNP Extension

This method creates a PNP Extension.

Table 639 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.01.16		subRC_SUCCESS subRC_APP_PNP_EXT_CODE_LEN_INVALID subRC_APP_PNP_EXT_CODE_INVALID subRC_APP_PNP_NAME_LEN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_PNP_EXT_CODE_LEN_INVALID subRC_APP_PNP_EXT_CODE_LEN_INVALID subRC_APP_PNP_EXT_CODE_TYPE_MISSING subRC_APP_PNP_EXT_CODE_TYPE_INVALID

Table 639 Create Pnp Extension Return Codes

Table 640 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
extPrefix	Represents a range of extension numbers or a single extension number with the extPrfx as the leading digits. The length of extPrfx must not be greater than extLen. Must be unique within the numbering plan.	integer	0..9	required
NumberPlanName	Name of the number plan.	1..31	Character string.	required
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
extLen	Extension number length.	integer	1..7	required
locCode	Consists of L2, L1, and L0 codes. The user interfaces will have the user enter the level codes in separate digit strings that the SOAP Server will concatenate into one digit string and derive the locL0Len, locL1Len, and locL2Len. • L1 and L2 are optional. • L1 cannot be administered without L0. • L2 cannot be administered without L1.	1..14	Numeric string.	0
e164Prefix	Represents the E.164 DN (national number) assigned to the range of extension numbers. This is to be provisioned if the extension numbers of the range have E.164 DNs assigned. It must be unique within the table.	1..15	Numeric string	“”
e164OfficeCode	Unused.			

Table 640 Create Pnp Extension Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
destType	Type of destination designated by the destId. The only valid value is HOMEDN(12). The destId must be an officeCodeId in the e164_office_code_t table. The e164Prfx must contain the officeCode as the leading substring. When e164Pfx is entered, the destType must be HOMEDN and the destId is to be the e164officeCodeId.	enum	PnpExt_HomeDn Note that the following additional values are defined but not currently supported: PnpExt_EndPoint, PnpExt_Line, PnpExt_Service, PnpExt_Vacant, PnpExt_Intercept	required
destName	Indicates the name of the destination.	1..15	Character string.	“”

Table 640 Create Pnp Extension Input Fields (Sheet 2 of 2)

12.5.2 Modify PNP Extension

This method modifies an existing PNP Extension.

Table 641 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.01.16		subRC_SUCCESS subRC_APP_PNP_EXT_CODE_LEN_INVALID subRC_APP_PNP_EXT_CODE_INVALID subRC_APP_PNP_NAME_LEN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_PNP_EXT_CODE_LEN_INVALID subRC_APP_PNP_EXT_CODE_LEN_INVALID subRC_APP_PNP_EXT_CODE_TYPE_MISSING subRC_APP_PNP_EXT_CODE_TYPE_INVALID subRC_APP_PNP_EXT_CODE_MODIFY_ERROR

Table 641 Modify Pnp Extension Return Codes

Table 642 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
extPrefix	Represents a range of extension numbers or a single extension number with the extPrfx as the leading digits. The length of extPrfx must not be greater than extLen. Must be unique within the numbering plan.	integer	0..9	required
NumberPlanName	Name of the number plan.	1..31	Character string.	required

Table 642 Modify Pnp Extension Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
extLen	Extension number length.	integer	1..7	required
locCode	Consists of L2, L1, and L0 codes. The user interfaces will have the user enter the level codes in separate digit strings that the SOAP Server will concatenate into one digit string and derive the locL0Len, locL1Len, and locL2Len. • L1 and L2 are optional. • L1 cannot be administered without L0. • L2 cannot be administered without L1.	1..14	Numeric string.	0
e164Prefix	Represents the E.164 DN (national number) assigned to the range of extension numbers. This is to be provisioned if the extension numbers of the range have E.164 DNs assigned. It must be unique within the table.	1..15	Numeric string	“”
e164OfficeCode	Unused.			
destType	Type of destination designated by the destId. The only valid value is HOMEDN(12). The destId must be an officeCodeId in the e164_office_code_t table. The e164Prfx must contain the officeCode as the leading substring. When e164Pfx is entered, the destType must be HOMEDN and the destId is to be the e164officeCodeId.	enum	PnpExt_HomeDn Note that the following additional values are defined but not currently supported: PnpExt_EndPoint, PnpExt_Line, PnpExt_Service, PnpExt_Vacant, PnpExt_Intercept	required
destName	Indicates the name of the destination.	1..15	Character string.	“”

Table 642 Modify Pnp Extension Input Fields (Sheet 2 of 2)

12.5.3 Delete PNP Extension

This method deletes the specified PNP Extension.

Table 643 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.01.16		subRC_SUCCESS subRC_APP_PNP_EXT_CODE_LEN_INVALID subRC_APP_PNP_EXT_CODE_INVALID subRC_APP_PNP_NAME_LEN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_PNP_EXT_CODE_DELETION_FAILED

Table 643 Delete Pnp Extension Return Codes

Table 644 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
extPrefix	Represents a range of extension numbers or a single extension number with the extPrfx as the leading digits. The length of extPrfx must not be greater than extLen. Must be unique within the numbering plan.	integer	0..9	required
NumberPlanName	Name of the number plan.	1..31	Character string.	required
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			

Table 644 Delete Pnp Extension Input Fields

12.5.4 Get PNP Extension List

This method retrieves a list of all PNP Extensions on the system.

Table 645 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.01.16		subRC_SUCCESS subRC_SYS_PNP_EXT_CODE_ALLOC_ERROR subRC_APP_PNP_EXT_CODE_LEN_INVALID subRC_APP_PNP_EXT_CODE_INVALID subRC_APP_PNP_NAME_LEN_INVALID subRC_APP_PNP_EXT_CODE_DOES_NOT_EXIST subRC_DB_PNP_EXT_CODE_LOOKUP_FAILED

Table 645 Get PnP Extension List Return Codes

Table 646 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
extPrefix	Represents a range of extension numbers or a single extension number with the extPrfx as the leading digits. The length of extPrfx must not be greater than extLen. Must be unique within the numbering plan.	integer	0..9	required
NumberPlanName	Name of the number plan.	1..31	Character string.	required
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			N/A

Table 646 Get PnP Extension List Input Fields

13 Number Modification

From the end user perspective, it is useful for the called party to see a number being displayed that can be used exactly to call back the calling party (instead of prefixing the public access code, country code, etc. to the display number before calling back). In order to support the display of a dialable number, there are administrative tables for number modification.

The dialable display number feature is per numbering plan based. It is applicable for all markets, all solutions. All these display number related tables are optional.

Any existing features or future features which have the number display functionality shall interact with the dialable display number feature. The dialable display number feature applies only if the display number supporting tables are provisioned. If they are not provisioned, then the current display number function will be used with no modification applied. To disable the dialable display number feature, the entries in these tables need to be removed.

The following subsections describe the SOAP/XML interfaces that exist for the provisioning of the Number Modification tables

13.1 DNM Prefixes

This table specifies the network access codes and prefixes that are defined for a specific numbering plan, a specific business group or system wide.

This table is consulted 2 ways:

- One way is to find out the type of number for an unknown digit string. It will find the prefixes table entry with the longest digit string match on network access code (PNAC) and prefix, strip the network access code and prefix and return the stripped number together with the type of number (TON) that is defined for that network access code and prefix.
- Another way is to find the PNAC and prefix digits for a specific type of number. It will find the number prefix entry for the specified numbering plan and business group.

13.1.1 Create DNM Prefix

Addition of DNM Prefix entries will be assisted through the use of the aforementioned web service. If BgName is empty, meaning the 'ANY' value, the NpName should be empty as well. The craft person will have to provide the corresponding Prefix attributes (see input field definitions below). Validation of a new DNM Prefix entry will ensure that all NPI/TON combinations in the list of data objects are unique. SOAP will support a series of validations regarding the input

parameters (presence of mandatory fields, usage of invalid characters, exceeding parameter length, invalid business group name etc). SOAP will communicate the outcome of the operation through the use of return codes.

Table 647 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.20.01	subRC_APP_NOT_ALLOWED_SETTING_KEYSET_TO_MOBILEDN subRC_APP_FEAT_OSMO_ONS_NOT_ASSIGNED_TO_DESK_PHONE_DN subRC_APP_FEAT_OSMO_DESK_PHONE_NUMBER_IN_USE subRC_APP_SUB_MOBL_RING_DUR_INVALID subRC_APP_SUB_MAIN_RING_DUR_INVALID subRC_APP_SUB_CELL_RING_DUR_INVALID subRC_APP_FEATURES_AND_OSMO_CONFLICT subRC_APP_FEATURE_PROFILE_OSMO_CONFLICT subRC_APP_SUB_OSMO_FEATURES_CONFLICT subRC_APP_SUB_OSMO_FP_CONFLICT subRC_APP_SUB_OSMO_CPG_MEMBER_CONFLICT subRC_APP_SUB_OSMO_COM_MEMBER_CONFLICT subRC_APP_SUB_OSMO_EA_MEMBER_CONFLICT subRC_APP_SUB_OSMO_MLHG_MEMBER_CONFLICT subRC_APP_SUB_OSMO_KEYSET_CONFLICT subRC_APP_SUB_OSMO_REMOVAL_FAILURE subRC_APP_OSMO_TLS_CONFLICT subRC_APP_FEAT_OSMO_CSTA_NOT_ASSIGNED_TO_DESK_PHONE_DN

Table 647 Create DNM Prefix Return Codes

Table 648 lists the input fields for this method.

Field Name	Required	Description	Length	Range	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
DnmPrefixKey	Yes	This structure has the mandatory fields which define uniquely in the OSCV a DNM Prefix entry. These fields are given below			
DnmPrefixDataList	No	This structure is a list containing instances that have all the other fields apart from the key that comprise a DNM Prefix entry. These fields are given below			
(DnmPrefixKey) BgName	Yes	The name of the Business Group. When the corresponding XML tag has no value (tag is empty) the ‘ANY’ BgName will be used instead.	Char array	30 (1…30)	N/A
(DnmPrefixKey) NpName	Yes	The name of the Originating Numbering Plan for this DN Prefix entry. Empty XML tag implies usage of NP ‘ANY’	Char array	30 (1…30)	N/A

Table 648 "Create DNM Prefix Input Fields (Sheet 1 of 2)

Field Name	Required	Description	Length	Range	Default Value
(DnmPrefixKey) TypeOfNumber	No	The type of Network Access Code and Prefix definitions	Enumerator	DNM_TON_International, DNM_TON_National, DNM_TON_Subscriber, DNM_TON_L2, DNM_TON_L1, DNM_TON_L0	DNM_TON_International
(DnmPrefixData) NetworkAccessCode	No	Describes the public or private network access digits.	Numeric string representing a valid dialable network access code.	15 (1…15)	Empty String
(DnmPrefixData) PrefixCode	No	The prefixed digits for the particular call type.	Numeric string representing a valid dialable prefix.	15 (1…15)	Empty String

Table 648 "Create DNM Prefix Input Fields (Sheet 2 of 2)

13.1.2 Modify DNM Prefix

Brief Service description: The craft person will be able to modify the Network Access Code and Prefix code of an already existing DNM Prefix entry using this specific web service. The craft person will still have to provide mandatory field/filters (see input field definitions below). SOAP will support a series of validations similar to the ones performed during the add operation. SOAP will communicate the outcome of the operation through the use of return codes which will differentiate depending on the case (see return code list below).

Table 649 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.01.03	subRC_APP_DSA_BLOCKED_BY_OP_MODE subRC_APP_CONGESTION_OVERLOAD subRC_APP_COMM_OPERATOR_INVALID subRC_APP_INSUFFICIENT_PERMISSIONS subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_NAME_LEN_INVALID subRC_APP_DNM_DEFINITION_NP_INVALID subRC_APP_DNM_PNAC_INVALID subRC_APP_DNM_PREFIX_INVALID subRC_APP_DNM_PREFIX_ALREADY_EXISTS subRC_APP_DNM_PREFIX_COULD_NOT_BE_CREATED subRC_DB_NDM_DATABASE_ERROR subRC_APP_NDM_ERROR subRC_SUCCESS

Table 649 Modify DNM Prefix Return Codes

Table 650 lists the input fields for this method.

Field Name	Required	Description	Length	Range	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
DnmPrefixKey	Yes	This structure has the mandatory fields which define uniquely in the OSCV a DNM Prefix entry. These fields are given in the Create DNM Prefix operation. None of these fields are modifiable			
DnmPrefixModKey	No	This structure has the mandatory fields which define uniquely in the OSCV a DNM Prefix entry. These fields are given in the Create DNM Prefix operation. These fields are modifiable			
DnmPrefixData	Yes	A list structure containing instances that have all the other fields apart from the key that comprise a DNM Prefix entry. These fields are given in the Create DNM Prefix description. Note: For all these fields, if the tag is not passed then the previous value of the field remains intact. Moreover, at least one of these fields need to exist in the structure.			

Table 650 Modify DNM Prefix Input Input Fields

13.1.3 Delete DNM Prefix

Brief Service description: The deletion of existing DNM Prefix entries will be assisted through the use of this web service. The craft person will have to provide the key fields/filters (see input field definitions below). SOAP will communicate the outcome of the removal through the use of different return codes depending on the case (see return code list below).

Table 651 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.01.03	subRC_APP_DSA_BLOCKED_BY_OP_MODE subRC_APP_CONGESTION_OVERLOAD subRC_APP_COMM_OPERATOR_INVALID subRC_APP_INSUFFICIENT_PERMISSIONS subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_NAME_LEN_INVALID subRC_APP_DNM_DEFINITION_NP_INVALID subRC_APP_DNM_PREFIX_WAS_NOT_FOUND subRC_APP_DNM_PREFIX_COULD_NOT_BE_DELETED subRC_DB_NDM_DATABASE_ERROR subRC_APP_NDM_ERROR subRC_SUCCESS

Table 651 Delete DNM Prefix Return Codes

Table 652 lists the input fields for this method.

Field Name	Required	Description	Length	Range	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
Transactional	No	This flag indicates whether processing of requested list of Definitions is to be treated as a transactional block where all entries are expected to succeed or fail as a whole. When set to True, atomic operations will be followed by a Commit/Rollback statement allowing them to be treated as a single block. Potential failed atomic operations are identified and displayed as part of the result of this API when Transactional field set to False	Boolean	[true|false]	false
DeleteDnmPrefixList	Yes	A list specifying all the entries to be deleted. At least one element should be given for deletion. Each element of the list is of type DnmPrefixKey described in the Create DNM Prefix operation			

Table 652 Delete DNM Prefix Input Fields

Table 653 lists the output fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.

Table 653 Delete DNM Prefix Output Fields(Sheet 1 of 2)

Field Name	Description
ErrorList	A structure that will be filled with the entries for which their deletion failed. If the operation was successful for all the entries, this structure will not be returned.

Table 653 Delete DNM Prefix Output Fields(Sheet 2 of 2)

13.1.4 Query DNM Prefix

Brief Service description: A list of DNM Prefixes will be included in the result of this web service. The craft person will have to provide a Business Group Name and/or a Number Plan name (see input field definitions below). SOAP will return the list of Prefix entries (if any) along with the outcome of the operation in case an error occurred (see return code list below). This operation will support paging and sorting (Order: ascending / descending - Type: by NDAL Weight / by Business Group / by Number Plan).

Table 654 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.01.03	subRC_APP_CONGESTION_OVERLOAD subRC_APP_COMM_OPERATOR_INVALID subRC_APP_INSUFFICIENT_PERMISSIONS subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_NAME_LEN_INVALID subRC_APP_DNM_DEFINITION_NP_INVALID subRC_APP_DNM_NDM_ERROR subRC_APP_DNM_PREFIX_NOT_FOUND subRC_SYS_SUB_FEATURE_NO_MEMORY subRC_SUCCESS

Table 654 Query DNM Prefix Return Codes

Table 655 lists the input fields for this method.

Field Name	Required	Description	Length	Range	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
DmmQueryOptions	No	A structure containing query options for all the DNM Library's objects			
BlockSystemWideEntries	No	Indicates whether system wide entries will also be part of the result list.	boolean	TRUE/FALSE	FALSE
(DmmQueryOptions) Paging	No	Optional Paging structure that will be used when part of the result is required			

Table 655 Query DNM Prefix Input Fields(Sheet 1 of 2)

Field Name	Required	Description	Length	Range	Default Value
(DmmQueryOptions) (Paging) PageSize	No	Page size of the retrieved result set	Unsigned Integer	[1-1000]	0 (No paging)
(DmmQueryOptions) (Paging) PageNumber	No	Page number of the retrieved result set	Unsigned Integer	[1-1000]	0 (No paging)
(DmmQueryOptions) SortType	No	The sort column of the retrieved result set	Enumerator	DNM_SortType_Weights, DNM_SortType_BgName, DNM_SortType_NpName,	DNM_SortType_Weights
(DmmQueryOptions) SortOrder	No	The sort order of the retrieved result set	Enumerator	Ascending Descending	Ascending
DnmPrefixKey	No	This structure has the mandatory fields which define uniquely in the OSCV a DNM Prefix entry. These fields are given in the Create DNM Prefix operation For the query operation all fields are optional. Empty BG tag implies ANY BG. No BG tag implies ALL BGs.			

Table 655 Query DNM Prefix Input Fields(Sheet 2 of 2)

Table 656 lists the output fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.
PaginatedInfo	A common paging structure containing Current Page/ Num of Pages / Page Size etc.
QueryDnmPrefixList	A list containing Prefix entries according to the provided criteria. List elements are instances consist of a DnmPrefixKey and a DnmPrefixData described in the DNM Create Prefix operation. Returned only in success.

Table 656 Query DNM Prefix Output Fields

13.1.5 Query DNM PrefixList

Brief Service description: A list of DNM Prefixes analyzed per type of number will be included in the result of this web service. The craft person will have to provide a Business Group Name and/or a Number Plan name (see input field definitions below). SOAP will return the list of Prefix entries (if any) along with the outcome of the operation in case an error occurred (see return code list below). This operation will support paging and sorting (Order: ascending / descending - Type: by NDAL Weight / by Business Group / by Number Plan)

Table 654 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.01.03	subRC_APP_CONGESTION_OVERLOAD subRC_APP_COMM_OPERATOR_INVALID subRC_APP_INSUFFICIENT_PERMISSIONS subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_NAME_LEN_INVALID subRC_APP_DNM_DEFINITION_NP_INVALID subRC_APP_DNM_NDM_ERROR subRC_APP_DNM_PREFIX_NOT_FOUND subRC_SYS_SUB_FEATURE_NO_MEMORY subRC_SUCCESS

Table 657 Query DNM PrefixList Return Codes

Table 658 lists the input fields for this method.

Field Name	Required	Description	Length	Range	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
DmmQueryOptions	No	A structure containing query options for all the DNM Library's objects			
BlockSystemWideEntries	No	Indicates whether system wide entries will also be part of the result list.	boolean	TRUE/FALSE	FALSE
(DmmQueryOptions) Paging	No	Optional Paging structure that will be used when part of the result is required			
(DmmQueryOptions) (Paging) PageSize	No	Page size of the retrieved result set	Unsigned Integer	[1-1000]	0 (No paging)
(DmmQueryOptions) (Paging) PageNumber	No	Page number of the retrieved result set	Unsigned Integer	[1-1000]	0 (No paging)
(DmmQueryOptions) SortType	No	The sort column of the retrieved result set	Enumerator	DNM_SortType_Weights, DNM_SortType_BgName, DNM_SortType_NpName,	DNM_SortType_Weights
(DmmQueryOptions) SortOrder	No	The sort order of the retrieved result set	Enumerator	Ascending Descending	Ascending
BgName	No	The Name of the Business Group			
NpName	No	The Name of the Number Plan			

Table 658 Query DNM PrefixList Input Fields

Table 659 lists the output fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.
PaginatedInfo	A common paging structure containing Current Page/ Num of Pages / Page Size etc.
QueryDnmPrefixList	A list containing enhanced Prefix entries according to the provided criteria. Returned only in success.

Table 659 Query DNM Prefix Output Fields

13.2 DNM Definitions

This table specifies the number definition entries that exist within a numbering plan, business group or even system wide.

The definitions table should only be populated with all the office codes that are created in the OpenScape Voice database. This includes public office codes and also office codes created for private Home DNs. It should never be necessary to repeat office code definitions; therefore, it is now also possible to create business group specific and even system wide definitions.

The 'old' definitions table also contained number normalization and number conversion entries. These entries have been moved to the Normalizations and Conversions tables.

The definitions table has now a new column to specify whether the definition is for a private or a public office code. This column is called the Numbering Plan Identifier (NPI). This is the same identifier that is also supported by ISDN and can have the values Public or Private.

Another new column in this table is the Local Toll column. This column allows the administrator specifying how public numbers are defined for users of this specific definition table entry. Per default it is assumed that users that have a specific public number actually follow the basic rules that they need to dial local numbers within their own area code, national numbers within their country and international numbers elsewhere. As long as these rules apply, there's no need for a local toll entry on the public number definition. However, if these rules change, as they certainly can in the American market, then the local toll table entry specified on the number definition will be used to find out how a number received from the public network can be optimized to the national or subscriber level for the subscribers of that office code. There are 2 situations where the local toll table is mandatory:

1. When a standalone private office code is created without a number conversion entry that allows the DNM library to find the accompanying public number then the local toll table entry will allow the DNM library to optimize incoming public numbers correctly for these users. E.g. if a branch office has a single access number and all branch office members have private extensions not reachable directly from the public network, then the

subscribers of the branch office need to be provisioned with private numbers and the private number definition table entry needs a local toll entry to define the local and national public numbers in the context of the private number subscribers.

2. When a public office code is created with subscribers that need to dial within split or overlapping area codes. This is the case in e.g. Boca Raton, FL in case the numbers presented to the subscribers need to be in the exact format as specified by the public network.

13.2.1 Create DNM Definition

Brief Service description: DNM Definition entries will be created using the aforementioned web service. The craft person will have to provide the corresponding Definition attributes (see input field definitions below). SOAP will support a series of validations regarding the input parameters (presence of mandatory fields, usage of invalid characters, exceeding parameter length, invalid business group name etc). However, SOAP will have no control over logical errors including cases where L2, L1 or L0 parts become mandatory as this will be a feature of the NDM component. SOAP will communicate the outcome of the operation through the use of return codes which will differentiate depending on the case (see return code list below).

Table 660 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.01.03	subRC_APP_CONGESTION_OVERLOAD subRC_APP_COMM_OPERATOR_INVALID subRC_APP_INSUFFICIENT_PERMISSIONS subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_NAME_LEN_INVALID subRC_APP_DNM_DEFINITION_NP_INVALID subRC_APP_DNM_CODE2_INVALID subRC_APP_DNM_CODE1_INVALID subRC_APP_DNM_CODE0_INVALID subRC_APP_DNM_SKIP_DIGITS_INVALID subRC_APP_DNM_MIN_DIGITS_INVALID subRC_APP_DNM_MAX_DIGITS_INVALID subRC_APP_DNM_LOCAL_TOLL_NAME_INVALID subRC_APP_DNM_DEFINITION_ALREADY_EXISTS subRC_APP_DNM_DEFINITION_COULD_NOT_BE_CREATED subRC_DB_NDM_DATABASE_ERROR subRC_APP_NDM_ERROR subRC_SUCCESS

Table 660 Create DNM Definition Return Codes

Table 661 lists the input fields for this method.

Field Name	Required	Description	Length	Range	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
DnmDefinitionKey	Yes	This structure has the mandatory fields which define uniquely in the OSCV a DNM Normalization entry. These fields are given below			
DnmDefinitionData	No	This structure has all the other fields apart from the key that comprise a DNM Normalization entry. These fields are given below			
(DnmDefinitionKey) BgName	Yes	The name of the Business Group. When the corresponding XML tag has no value (tag is empty) the 'ANY' BgName will be used instead.	Char array	30 (1…30)	N/A
(DnmDefinitionKey) NpName	Yes	The name of the Originating Numbering Plan for this DN Definition entry. Empty XML tag implies usage of NP 'ANY'.	Char array	30 (1…30)	N/A
(DnmDefinitionKey) Npi	Yes	The type of numbering plan format to apply.	Enumerato r	DNM_NPI_Public, DNM_NPI_Private	DNM_NPI_Public
(DnmDefinitionKey) Code2	Yes	Represents a Country code when the NP indicator is set to 'Public' or a Level 2 code when the NP indicator is set to 'Private'.	Char array	15 (1…15)	N/A
(DnmDefinitionKey) Code1	Yes	Represents an Area code when the NP indicator is set to 'Public' or a Level 1 code when the NP indicator is set to 'Private'.	Char array	15 (1…15)	N/A
(DnmDefinitionKey) Code0	Yes	Represents a Local office code when the NP indicator is set to 'Public' or a Level 0 code when the NP indicator is set to 'Private'.	Char array	15 (1…15)	N/A
(DnmDefinitionData) SkipDigit	No	Skip Digit value for the Definition entry. Specifies the number of digits which are to be skipped to extract the Local Office/L0 code when extension number overlaps.	Integer	[1-15]	0
(DnmDefinitionData) MinDigits	No	Minimum digits of the defined number.	Integer	Within the ranges of the definition	-1
(DnmDefinitionData) MaxDigits	No	Maximum digits of the defined number.	Integer	Within the ranges of the definition	-1
(DnmDefinitionData) LocalTollName	No	The name of Local Toll table.	Char array	30 (0…30)	Empty String

Table 661 Create DNM Definition Input Fields

Table 662 lists the output fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.

Table 662 Create DNM Definition Output Fields

13.2.2 Modify DNM Definition

Brief Service description: DNM Definition entries will be modified using the aforementioned web service. The craft person will have to provide a number of filters (see input field definitions below). SOAP will support a series of validations regarding the input parameters (presence of mandatory fields, usage of invalid characters, exceeding parameter length, invalid business group name etc). However, SOAP will have no control over logical errors including cases where L2, L1 or L0 parts become mandatory as this will be a feature of the NDM component. SOAP will communicate the outcome of the operation through the use of return codes which will differentiate depending on the case (see return code list below).

Table 663 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.01.03	subRC_APP_DSA_BLOCKED_BY_OP_MODE subRC_APP_CONGESTION_OVERLOAD subRC_APP_COMM_OPERATOR_INVALID subRC_APP_INSUFFICIENT_PERMISSIONS subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_NAME_LEN_INVALID subRC_APP_DNM_DEFINITION_NP_INVALID subRC_APP_DNM_CODE2_INVALID subRC_APP_DNM_CODE1_INVALID subRC_APP_DNM_CODE0_INVALID subRC_APP_DNM_SKIP_DIGITS_INVALID subRC_APP_DNM_MIN_DIGITS_INVALID subRC_APP_DNM_MAX_DIGITS_INVALID subRC_APP_DNM_LOCAL_TOLL_NAME_INVALID subRC_APP_DNM_DEFINITION_WAS_NOT_FOUND subRC_APP_DNM_DEFINITION_COULD_NOT_BE_DELETED subRC_DB_NDM_DATABASE_ERROR subRC_APP_NDM_ERROR subRC_SUCCESS

Table 663 Modify DNM Definition Return Codes

Table 664 lists the input fields for this method.

Field Name	Required	Description	Length	Range	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
DnmDefinitionKey	Yes	This structure has the mandatory fields which define uniquely in the OSCV a DNM Definition entry. These fields are given in the Create DNM Definition operation. All but BgName, NpName fields are modifiable			
DnmDefinitionModKey	No	This structure has the modified set of values for the mandatory fields which define uniquely in the OSCV a DNM Definition entry and are partially modifiable.			
DnmDefinitionData	Yes	This structure has all the other fields apart from the key that comprise a DNM Definition entry. These fields are given in the Create DNM Definition description. Note: The optional fields, if omitted will remain intact			

Table 664 Modify DNM Definition Input Fields

Table 665 lists the output fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.

Table 665 Modify DNM Definition Output Fields

13.2.3 Delete DNM Definition

Brief Service description: The deletion of existing DNM Definition entries will be assisted through the use of this web service. The craft person will have to provide the key fields/filters (see input field definitions below). SOAP will communicate the outcome of the removal through the use of different return codes depending on the case (see return code list below).

Table 666 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.01.03	subRC_APP_DSA_BLOCKED_BY_OP_MODE subRC_APP_CONGESTION_OVERLOAD subRC_APP_COMM_OPERATOR_INVALID subRC_APP_INSUFFICIENT_PERMISSIONS subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_NAME_LEN_INVALID subRC_APP_DNM_DEFINITION_NP_INVALID subRC_APP_DNM_CODE2_INVALID subRC_APP_DNM_CODE1_INVALID subRC_APP_DNM_CODE0_INVALID subRC_APP_DNM_SKIP_DIGITS_INVALID subRC_APP_DNM_MIN_DIGITS_INVALID subRC_APP_DNM_MAX_DIGITS_INVALID subRC_APP_DNM_LOCAL_TOLL_NAME_INVALID subRC_APP_DNM_DEFINITION_WAS_NOT_FOUND subRC_APP_DNM_DEFINITION_COULD_NOT_BE_DELETED subRC_DB_NDM_DATABASE_ERROR subRC_APP_NDM_ERROR subRC_SUCCESS

Table 666 Delete DNM Definition Return Codes

Table 667 lists the input fields for this method.

Field Name	Required	Description	Length	Range	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
Transactional	No	This flag indicates whether processing of requested list of Definitions is to be treated as a transactional block where all entries are expected to succeed or fail as a whole. When set to True, atomic operations will be followed by a Commit/Rollback statement allowing them to be treated as a single block. Potential failed atomic operations are identified and displayed as part of the result of this API when Transactional field set to False	Boolean	[true|false]	false
DeleteDnmDefinitionList	Yes	A list specifying all the entries to be deleted. Each element of the list is of type DnmDefinitionKey described in the Create DNM DefinitionKey operation. At least one element should be specified in the request			

Table 667 Delete DNM Definition Input Fields

Table 668 lists the output fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.
ErrorList	A structure that will be filled with the entries for which their deletion failed. If the operation was successful for all the entries, this structure will not be returned.

Table 668 Delete DNM Definition Output Fields

13.2.4 Query DNM Definition

Brief Service description: A list of DNM Definitions will be included in the result of this web service. The craft person will have to provide a Business Group Name and/or a Number Plan name (see input field definitions below). SOAP will return the list of Definition entries (if any) along with the outcome of the operation in case an error occurred (see return code list below). This operation will support paging and sorting (Order: ascending / descending - Type: by NDAL Weight / by Business Group / by Number Plan).

Table 669 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.01.03	subRC_APP_CONGESTION_OVERLOAD subRC_APP_COMM_OPERATOR_INVALID subRC_APP_INSUFFICIENT_PERMISSIONS subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_NAME_LEN_INVALID subRC_APP_DNM_DEFINITION_NP_INVALID subRC_APP_DNM_NDM_ERROR subRC_APP_DNM_DEFINITION_NOT_FOUND subRC_SYS_SUB_FEATURE_NO_MEMORY subRC_SUCCESS

Table 669 Query DNM Definition Return Codes

Table 670 lists the input fields for this method.

Field Name	Required	Description	Length	Range	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
BlockSystemWide Entries	No	Indicates whether system wide entries will also be part of the result list.	boolean	TRUE/FALSE	FALSE
DmmQueryOption s	No	A structure containing query options for all the DNM Library's objects			

Table 670 Query DNM Definition Input Fields(Sheet 1 of 2)

Field Name	Required	Description	Length	Range	Default Value
(DmmQueryOptions) Paging	No	Optional Paging structure that will be used when part of the result is required			
(DmmQueryOptions) (Paging) PageSize	No	Page size of the retrieved result set	Unsigned Integer	[1-1000]	0 (No paging)
(DmmQueryOptions) (Paging) PageNumber	No	Page number of the retrieved result set	Unsigned Integer	[1-1000]	0 (No paging)
(DmmQueryOptions) SortType	No	The sort column of the retrieved result set	Enumerator	DNM_SortType_Weights, DNM_SortType_BgName, DNM_SortType_NpName,	DNM_SortType_Weights
(DmmQueryOptions) SortOrder	No	The sort order of the retrieved result set	Enumerator	Ascending Descending	Ascending
DnmDefinitionKey	No	This structure has the mandatory fields which define uniquely in the OSCV a DNM Definition entry. These fields are given in the Create DNM Definition operation For the query operations all fields are optional. Empty BG tag implies ANY BG. No BG tag implies all BGs. If any of the fields is left blank all matching values will be returned. E.g. if npi = public and CC=1 then all public definitions for the USA will be returned in a list.			

Table 670 Query DNM Definition Input Fields(Sheet 2 of 2)

Table 671 lists the output fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.
PaginatedInfo	A common paging structure containing Current Page/ Num of Pages / Page Size etc.
QueryDnmDefinition List	A list containing the actual Definition objects stored in DB. Each item of the list is of type QueryDnmDefinitionEntry which consists of a part of type DNMDefinitionKey and another of type DNMDefinitionData and the index of the entry stored in DB. Returned only in success.

Table 671 Query DNM Definition Output Fields

13.2.5 Query DNM Definition

Brief Service description: A list of DNM Definitions will be included in the result of this web service. The craft person will have to provide a Business Group Name and/or a Number Plan name (see input field definitions below). SOAP will return the list of Definition entries (if any) along with the outcome of the operation in case

an error occurred (see return code list below). This operation will support paging and sorting (Order: ascending / descending - Type: by NDAL Weight / by Business Group / by Number Plan).

Table 672 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.01.03	subRC_APP_CONGESTION_OVERLOAD subRC_APP_COMM_OPERATOR_INVALID subRC_APP_INSUFFICIENT_PERMISSIONS subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_NAME_LEN_INVALID subRC_APP_DNM_DEFINITION_NP_INVALID subRC_APP_DNM_NDM_ERROR subRC_APP_DNM_DEFINITION_NOT_FOUND subRC_SYS_SUB_FEATURE_NO_MEMORY subRC_SUCCESS

Table 672 Query DNM Definition Return Codes

Table 673 lists the input fields for this method.

Field Name	Required	Description	Length	Range	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
BlockSystemWide Entries	No	Indicates whether system wide entries will also be part of the result list.	boolean	TRUE/FALSE	FALSE
DmmQueryOption s	No	A structure containing query options for all the DNM Library's objects			
(DmmQueryOptio ns) Paging	No	Optional Paging structure that will be used when part of the result is required			
(DmmQueryOptio ns) (Paging) PageSize	No	Page size of the retrieved result set	Unsigned Integer	[1-1000]	0 (No paging)
(DmmQueryOptio ns) (Paging) PageNumber	No	Page number of the retrieved result set	Unsigned Integer	[1-1000]	0 (No paging)
(DmmQueryOptio ns) SortType	No	The sort column of the retrieved result set	Enumerato r	DNM_SortType_Weig hts, DNM_SortType_BgNa me, DNM_SortType_NpNa me,	DNM_SortType_W eights

Table 673 Query DNM Definition Input Fields(Sheet 1 of 2)

Field Name	Required	Description	Length	Range	Default Value
(DmmQueryOptions) SortOrder	No	The sort order of the retrieved result set	Enumerator	Ascending Descending	Ascending
DnmDefinitionKey	No	This structure has the mandatory fields which define uniquely in the OSCV a DNM Definition entry. These fields are given in the Create DNM Definition operation For the query operations all fields are optional. Empty BG tag implies ANY BG. No BG tag implies all BGs. If any of the fields is left blank all matching values will be returned. E.g. if npi = public and CC=1 then all public definitions for the USA will be returned in a list.			

Table 673 Query DNM Definition Input Fields(Sheet 2 of 2)

Table 674 lists the output fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.
PaginatedInfo	A common paging structure containing Current Page/ Num of Pages / Page Size etc.
QueryDnmDefinition List	A list containing the actual Definition objects stored in DB. Each item of the list is of type QueryDnmDefinitionEntry which consists of a part of type DNMDefinitionKey and another of type DNMDefinitionData and the index of the entry stored in DB. Returned only in success.

Table 674 Query DNM Definition Output Fields

13.3 DNM Modification

This table specifies the number modification rules that are applied for creating a dialable number.

The administrator gets full flexibility on specifying very specific rules that only apply for number modification from a specific numbering plan to a specific endpoint or very general rules that apply system wide.

Other changes include specification of:

- Allow specifying a rule that applies to all Private Networking endpoints (ALL-PN) or to all endpoints (ALL) of a specific numbering plan, a specific business group or system wide. In order to allow ALL-PN to be specified there must be at least one Private Networking (SIP or SIP-Q) endpoint configured with the specified numbering plan, business group or system wide. In order to allow ALL to be specified there must be at least one non-subscriber endpoint configured with the specified numbering plan, business group or system wide.
- Use of a Prefix flag on a rule instead of specifying it on the output type of number field.

- Use of a Presentation Restricted flag on a rule instead of specifying it in the Number Transmission field.
- Renaming the Number Transmission field to Number Source field with 3 possible values: Input Number (formerly known as Transparent or Blocked), Default Home DN and BG Display Number.
- Use of a Default Home DN number provisioned on the specified endpoint instead of the provided input number. This only succeeds if the dialable number is created for an endpoint that has a Default Home DN number defined. This is done by selecting the Default Home DN Number Source. It will not be allowed to specify a Default Home DN rule without having either the specified endpoint or one of the private networking and/or trunking endpoints (ALL-PN/ALL) in the numbering plan with a defined Default Home DN.
- The ALL input type of number is not used anymore. Instead up to 3 ANY rules (prioritized 1 to 3) can be specified that take an output type of number that is not ANY. The output type of number ANY automatically receives reserved priority number 4. It shall not be possible to enter 2 modification rules with the same number source field value that only differ on priority. E.g. an ANY (input) / INT (output) / 1 (priority) / Input Number (Number Source) rule precludes creating an ANY (input) / INT (output) / 2 (priority) or 3 (priority) / Input Number (Number Source) rule. The priorities 1 to 4 will also apply to the rules with a defined input type of number.
- Maximum optimization level that applies to a modification rule. This replaces the previous rules that just allowed optimization one level down or optimization all the way to an extension.
- Fallback rules that apply when a higher priority rule fails. The number modification library does not necessarily fail anymore when the first matching rule does not deliver a result. Instead the next best rule is searched and applied until either no rules are available anymore or a rule ends up into a positive modification.

There are a few additional output types of numbers. For all the following rules, Optimize Type of Number must be None and the Prefix Required flag must not be set (both fields are to be grayed out by the OpenScape Voice Assistant):

- FQPN: request to modify the number to a fully qualified private number (L2, L1 or L0 number).
- EXT: request to modify the number to an extension number.
- NORMALIZED: request to return the fully qualified number for the input number. This is a fully qualified private number if the input number is a private number or a fully qualified public number if the input number is a public number.

For inter business group rules (where 2 different business group names are specified), the output type of number must be a public type of number: one of Any, International, National, Subscriber or Normalized. Inter business group rules with non-public number outputs will be skipped.

13.3.1 Create DNM Modification

Brief Service description: Creation of DNM Modification entries will be accommodated by this specific web service. The craft person will have to provide the Modification attributes according to the input field definitions presented below. SOAP will support a series of validations regarding the input parameters (presence of mandatory fields, usage of invalid characters, exceeding parameter length, invalid business group name etc). The outcome of the operation will be communicated through the use of different return codes depending on the case (see return code list below).

Table 675 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.01.03	subRC_APP_CONGESTION_OVERLOAD subRC_APP_COMM_OPERATOR_INVALID subRC_APP_INSUFFICIENT_PERMISSIONS subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_NAME_LEN_INVALID subRC_APP_DNM_DEFINITION_NP_INVALID subRC_APP_DNM_CODE2_INVALID subRC_APP_DNM_CODE1_INVALID subRC_APP_DNM_CODE0_INVALID subRC_APP_DNM_SKIP_DIGITS_INVALID subRC_APP_DNM_MIN_DIGITS_INVALID subRC_APP_DNM_MAX_DIGITS_INVALID subRC_APP_DNM_LOCAL_TOLL_NAME_INVALID subRC_APP_DNM_DEFINITION_ALREADY_EXISTS subRC_APP_DNM_DEFINITION_COULD_NOT_BE_CREATED subRC_DB_NDM_DATABASE_ERROR subRC_APP_NDM_ERROR subRC_SUCCESS

Table 675 Create DNM Modification Return Codes

Table 676 lists the input fields for this method.

Field Name	Required	Description	Length	Range	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
DnmModificationKey	Yes	This structure has the mandatory fields which define uniquely in the OSCV a DNM Modification entry. These fields are given below			

Table 676 Create DNM Modification Input Fields(Sheet 1 of 3)

Field Name	Required	Description	Length	Range	Default Value
DnmModificationData	Yes	This structure has all the other fields apart from the key that comprise a DNM Modification entry. These fields are given below			
(DnmModification Key) FromBgName	Yes	The name of the Originating Business Group. When the corresponding XML tag has no value (tag is empty) the 'ANY' BgName will be used instead.	Char array	30 (1…30)	N/A
(DnmModification Key) ToBgName	Yes	The name of the Terminating Business Group. When the corresponding XML tag has no value (tag is empty) the 'ANY' BgName will be used instead.	Char array	30 (1…30)	N/A
(DnmModification Key) FromNpName	Yes	The name of the Originating Numbering Plan for this DN Modification entry. Empty XML tag implies usage of NP 'ANY'.	Char array	30 (1…30)	N/A
(DnmModification Key) ToNpName	Yes	The name of the Terminating Numbering Plan for this DN Modification entry. Empty XML tag implies usage of NP 'ANY'.	Char array	30 (1…30)	N/A
(DnmModification Key) ToEndpoint	Yes	A struct of type DnmEndpoint specifying the type of the Terminating Endpoint.			
(DnmModification Key) (DnmEndpoint) EnpointType	No	The type of the terminating endpoint	Enumerator	DNM_EpType_None DNM_EpType_ALL DNM_EpType_ALL_PN DNM_EpType_EpName	DNM_EpType_None
(DnmModification Key) (DnmEndpoint) EpName	No	The name of the Endpoint that this entry belongs.	Char array	16 (0…16)	Empty string
(DnmModification Key) InputTON	Yes	A part of the key used to qualify the type of number for which this Number Modification entry applies.	Enumerator	DNM_TON_nternational, DNM_TON_National, DNM_TON_Subscriber, DNM_TON_LocationLevel2, DNM_TON_LocationLevel1, DNM_TON_LocalSubscriber, DNM_ TON_Any	DNM_TON_International
(DnmModification Key) Priority	Yes	The value of priority that applies to the Input TON.	Unsigned Integer	[1-4]	1

Table 676 Create DNM Modification Input Fields(Sheet 2 of 3)

Field Name	Required	Description	Length	Range	Default Value
(DnmModification Data) OutputTON	Yes	Defines the conversion of an explicit “calling number” to a desired format before the number is transmitted to the destination.	Enumerator	DNM_TON_Unknown, DNM_ TON_International, DNM_ TON_National, DNM_ TON_Subscriber, DNM_ TON_LocationLevel2, DNM_ TON_LocationLevel1, DNM_ TON_LocalSubscriber, DNM_ TON_Any, DNM_ TON_FQPN, DNM_ TON_Normalized DNM_ TON_Extension	DNM_ TON_International
(DnmModification Data) NumberSource	No	Defines which number shall be taken as number source for this DNM Modification rule.	Enumerator	DNM_NS_InputNumber, DNM_ NS_BGDisplayNumber, DNM_ NS_DefaultHomeDN	DNM_NS_InputNumber
(DnmModification Data) OptimizeTON	No	The optimization rule for this DNM Modification entry.	Enumerator	DNM_ TON_None, DNM_ TON_International, DNM_ TON_National, DNM_ TON_Subscriber, DNM_ TON_LocationLevel2, DNM_ TON_LocationLevel1, DNM_ TON_LocationLevel0, DNM_ TON_Extension	DNM_TON_None
(DnmModification Data) RestrictPresentation	No	Defines whether presentation should be restricted.	Boolean	[True | False]	False
(DnmModification Data) PrefixRequired	No	Defines whether a prefixed number is to be returned.	Boolean	[True | False]	False

Table 676 Create DNM Modification Input Fields(Sheet 3 of 3)

Table 677 lists the output fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.

Table 677 Create DNM Modification Output Fields

13.3.2 Modify DNM Modification

Brief Service description: Modification of DNM Modification entries will be assisted by this specific web service. The craft person will have to provide the Modification attributes according to the input field definitions presented below. SOAP will support a series of validations regarding the input parameters (presence of mandatory fields, usage of invalid characters, exceeding parameter length, invalid business group name etc). The result of the operation will be communicated through the use of different return codes depending on the case (see return code list below).

Table 678 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.01.03	subRC_APP_DSA_BLOCKED_BY_OP_MODE subRC_APP_CONGESTION_OVERLOAD subRC_APP_COMM_OPERATOR_INVALID subRC_APP_INSUFFICIENT_PERMISSIONS subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_NAME_LEN_INVALID subRC_APP_DNM_DEFINITION_NP_INVALID subRC_APP_DNM_ENDPOINT_INVALID subRC_APP_DNM_TYPE_OF_NUMBER_INVALID subRC_APP_DNM_PRIORITY_INVALID subRC_APP_DNM_NUMBER_SOURCE_INVALID subRC_APP_DNM_MODIFICATION_ALREADY_EXISTS subRC_APP_DNM_MODIFICATION_COULD_NOT_BE_CREATED subRC_DB_NDM_DATABASE_ERROR subRC_APP_NDM_ERROR subRC_SUCCESS

Table 678 Modify DNM Modification Return Codes

Table 679 lists the input fields for this method.

Field Name	Required	Description	Length	Range	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	Required
DnmModificationKey	Yes	This structure has the mandatory fields which define uniquely in the OSCV a DNM Modification entry. These fields are given in the Create DNM Modification operation.			

Table 679 Modify DNM Modification Input Fields(Sheet 1 of 2)

Field Name	Required	Description	Length	Range	Default Value
DnmModificationModKey	No	This structure has the modified set of values for the mandatory fields which define uniquely in the OSCV a DNM Modification entry. Besides fields “Input Ton” and “Priority” everything else is non modifiable.			
DnmModificationData	Yes	This structure has all the other fields apart from the key that comprise a DNM Modification entry. These fields are given in the Create DNM Modification description. Note: The optional fields, if omitted, will remain intact. At least one field of this structure should exist in the request.			

Table 679 Modify DNM Modification Input Fields(Sheet 2 of 2)

Table 680 lists the output fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.

Table 680 Modify DNM Modification Output Fields

13.3.3 Delete DNM Modification

Brief Service description: The deletion of existing DNM Modification entries will be assisted through the use of this web service. The craft person will have to provide the key fields/filters (see input field definitions below). The output of this API will contain a proper return code indicating whether the operation resulted in success or failure (see return code list below).

Table 681 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.01.03	subRC_APP_DSA_BLOCKED_BY_OP_MODE subRC_APP_CONGESTION_OVERLOAD subRC_APP_COMM_OPERATOR_INVALID subRC_APP_INSUFFICIENT_PERMISSIONS subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_NAME_LEN_INVALID subRC_APP_DNM_DEFINITION_NP_INVALID subRC_APP_DNM_ENDPOINT_INVALID subRC_APP_DNM_TYPE_OF_NUMBER_INVALID subRC_APP_DNM_PRIORITY_INVALID subRC_APP_DNM_NUMBER_SOURCE_INVALID subRC_APP_DNM_MODIFICATION_WAS_NOT_FOUND subRC_APP_DNM_MODIFICATION_COULD_NOT_BE_REMOVED subRC_DB_NDM_DATABASE_ERROR subRC_APP_NDM_ERROR subRC_SUCCESS

Table 681 Delete DNM Modification Return Codes

Table 682 lists the input fields for this method.

Field Name	Required	Description	Length	Range	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
Transactional	No	This flag indicates whether processing of requested list of Definitions is to be treated as a transactional block where all entries are expected to succeed or fail as a whole. When set to True, atomic operations will be followed by a Commit/Rollback statement allowing them to be treated as a single block. Potential failed atomic operations are identified and displayed as part of the result of this API when Transactional field set to False	Boolean	[true|false]	false
DeleteDnmModificationList	No	A list specifying all the entries to be deleted. Each element of the list is of type DnmModificationKey described in the Create DNM Modification operation			

Table 682 Delete DNM Modification Input Fields

Table 683 lists the output fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.

Table 683 Delete DNM Modification Output Fields

13.3.4 Query DNM Modification

Brief Service description: A list of DNM Modification will be included in the result of this web service. The craft person will be able to provide a list of Business Groups that an OSV Administrator has access to. This will allow for a BG-restricted view of the DNM Modifications. SOAP will return the list of Modification entries along with the result of the operation (see return code list below). This operation will support paging and sorting (Order: ascending / descending - Type: by NDAL Weight / by Originating BG -NP / by Terminating BG - NP - EP).

Table 684 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.01.03	subRC_APP_DSA_BLOCKED_BY_OP_MODE subRC_APP_CONGESTION_OVERLOAD subRC_APP_COMM_OPERATOR_INVALID subRC_APP_INSUFFICIENT_PERMISSIONS subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_NAME_LEN_INVALID subRC_APP_DNM_DEFINITION_NP_INVALID subRC_APP_DNM_ENDPOINT_INVALID subRC_APP_DNM_TYPE_OF_NUMBER_INVALID subRC_APP_DNM_PRIORITY_INVALID subRC_APP_DNM_NUMBER_SOURCE_INVALID subRC_APP_DNM_MODIFICATION_WAS_NOT_FOUND subRC_APP_DNM_MODIFICATION_COULD_NOT_BE_REMOVED subRC_DB_NDM_DATABASE_ERROR subRC_APP_NDM_ERROR subRC_SUCCESS

Table 684 Query DNM Modification Return Codes

Table 685 lists the input fields for this method.

Field Name	Required	Description	Length	Range	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
BlockSystemWide Entries	No	Indicates whether system wide entries will be included in the result list.	Boolean	TRUE / FALSE	FALSE
DmmQueryOption s	No	A structure containing query options for all the DNM Library's objects			
(DmmQueryOptio ns) Paging	No	Optional Paging structure that will be used when part of the result is required			
(DmmQueryOptio ns) (Paging) PageSize	No	Page size of the retrieved result set	Unsigned Integer	[1-1000]	0 (No paging)
(DmmQueryOptio ns) (Paging) PageNumber	No	Page number of the retrieved result set	Unsigned Integer	[1-1000]	0 (No paging)

Table 685 Query DNM Modification Input Fields(Sheet 1 of 2)

Field Name	Required	Description	Length	Range	Default Value
(DmmQueryOptions) SortType	No	The sort column of the retrieved result set	Enumerator	DNM_SortType_NoSorting, DNM_SortType_Weights, DNM_SortType_BgName, DNM_SortType_NpName,	DNM_SortType_Weights
(DmmQueryOptions) SortOrder	No	The sort order of the retrieved result set	Enumerator	Ascending Descending	Ascending
DnmBgAccessList	No	A list of Business Groups that the Administrator has access to. Unauthorized BG / NP entries in the originating context will be obfuscated. Each list element contains a BGName attribute having the same characteristics as the other BG attributes in the request.			
DnmModificationKey	No	This structure has the mandatory fields which define uniquely in the OSCV a DNM Modification entry. These fields are given in the Create DNM Modification operation For the query operation all fields are optional. Empty BG tags implies ANY BG. No BG tag implies all BGs.			

Table 685 Query DNM Modification Input Fields(Sheet 2 of 2)

Table 686 lists the output fields for this method.

Field Name	Description
ResultCodeStruct	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.
PaginatedInfo	A common paging structure containing Current Page/ Num of Pages / Page Size etc.
QueryDnmModificationList	A list containing the actual Normalization objects stored in DB. Each item of the list is of type QueryDnmModificationElement which consists of a part of type DnmModificationKey and another of type DnmModificationData. Returned only in success

Table 686 Query DNM Modification Output Fields

13.4 DNM Normalizations

This is a new table that specifies the entries to normalize any type of input number to a fully qualified public (international) or private (L2, L1 or L0) number.

The administrator can specify endpoint specific, numbering plan specific, business group wide and even system wide entries. The purpose of this table is to normalize the numbers received from gateways and other trunking endpoints. Therefore, most likely, the administrator will be entering endpoint specific entries in this table. This table and the local toll table are the only tables that require knowledge beyond the boundaries of the OpenScape Voice system being administered. This table contains information about the format of the presentation numbers sent by specific endpoints. Note that this table is consulted after the

prefixes table has been consulted, so it usually works on defined numbers (not unknown) unless the PNAC and prefixes are not added by the gateway, at which point this table requires normalization entries for Unknown type of numbers.

This table allows specifying an Input Pattern and Type of Number used for matching the number to be normalized and then specifying a modification rule and the resulting type of number of the normalized number. The input pattern allows using wildcards and regular expressions and contains information on how to split the matching digit string in fields that can then be repeated in the modification rule.

This table is also used when a translation results in a call to a specific gateway. If the result of the translation (output of prefix access codes table) is not a fully qualified number then the Normalization table is used to try normalizing the number for presentation purposes in case the gateway does not provide a busy, alerting or connected number.

13.4.1 Input Patterns

In order to limit and even hide the complexity of regular expressions, the input patterns use the following wildcard syntax for pattern elements:

Wildcard	Equivalent element
N	[2-9]
X	[0-9]
Z	0 or more X's
Z{n,m}	Between n and m X's; n must be smaller than m.
Z{n}	Exactly n X's

Table 687 Pattern Wildcards

Furthermore, it will not be allowed to enter patterns that would result in regular expressions that would invoke backtracking. This means that the "|" delimiter used to indicate a choice of elements in regular expressions is forbidden from use in the input patterns.

The only other pattern elements allowed are:

- [nm]: n or m with n and m any digit between 0 and 9, *, #.
- [n-m]: n through m with n and m a digit between 0 and 9. It can be combined with the previous [nm] regular expression. E.g. [04-79] means 0, 4, 5, 6, 7 or 9.
- n: a digit between 0 and 9, * or #.

Pattern elements are joined in pattern fields through the use of the '-' delimiter between 2 pattern elements. Note that the '-' is also allowed inside a pattern element's square brackets and will then not be counted as field numbers.

The maximum length of an input pattern shall be 64 characters.

Examples of input patterns are:

Pattern	Description
8011-Z	8011 followed by an open-ended number of digits. Split in 2 fields: Field1 is 8011. Field2 contains the remaining digits of the matching input string.
81-Z{10}	81 followed by a fixed length 10 digit number. Split in 2 fields: Field1 is 81. Field2 contains a 10 digit number.
00-Z{10,11}	00 followed by a variable length 10 or 11 digit number. Split in 2 fields: Field1 is 00. Field2 contains a 10 or 11 digit number.
XXXX	A 4 digit number in a single field.

Table 688 DNM Definition Pattern Examples

If multiple entries could apply to the same input digit string, a weighting factor attributed to the wildcards ensures that specific digits have higher weight then N, followed by X and lastly Z wildcards. In the example in Figure 2 7 a 7-digit input digit string matches both the first (NXX-XXXX) and the last entry (Z). Because of the weighting, the NXX-XXXX entry will be the first matching entry.

13.4.2 Output Expression

An output expression is a series of elements and fields where the fields reference the matching digits of the corresponding field in the input pattern. The '-' delimiter is allowed anywhere in the output expression and is ignored in the resulting element string.

The maximum length of an output expression shall be 64 characters.

Examples of output expressions with the previous examples would be:

Pattern	Expression	Description
8011-Z	{2}	Return contents of field 2.
81-Z{10}	1{2}	Return ‘1’ followed by the contents of field 2. This would create a US international number from a prefixed national US number.

Pattern	Expression	Description
00-Z{10,11}	32{2}	Return '32' followed by the contents of field 2. This would create a BE international number from a prefixed national BE number.
XXXX	1561555-{1}	Return 1561555 followed by the contents of field 1 (the '-' is ignored). This would create a US international number from a 4 digit extension.

Table 689 DNM Definition Expression Examples

The following web services implemented from the OSCV soapServer component accommodate this new interface's functionality. They are described in following sections.

Web Service	Description
unsp__CreateDnmNormalization	Allows the creation of a new DNM Normalization entry
unsp__ModifyDnmNormalization	Allows the modification of an existing DNM Normalization entry
unsp__DeleteDnmNormalization	Allows the deletion of one or more (list) DNM Normalization entries
unsp__QueryDnmNormalization	Allows the retrieval of one or more (list) DNM Normalization entries

13.4.3 Create DNM Normalization

Brief Service description: Using this web service, the craft person will be able to create an entry in the display_number_normalization_t table of DB through NDM. The only optional fields when triggering this interface are the input/output TONs, which, if not passed in the request will have the default values of DNM_TON_Unknown and DNM_TON_International respectively.

Table 690 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.01.03	subRC_APP_DSA_BLOCKED_BY_OP_MODE subRC_APP_CONGESTION_OVERLOAD subRC_APP_INSUFFICIENT_PERMISSIONS subRC_APP_COMM_OPERATOR_INVALID subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_NAME_LEN_INVALID subRC_APP_DNM_NP_INVALID subRC_APP_DNM_EP_INVALID subRC_APP_DNM_INPUT_PATTERN_INVALID subRC_APP_DNM_INPUT_TON_INVALID subRC_APP_DNM_NORMALIZED_TON_INVALID subRC_APP_DNM_OUTPUT_EXPRESSION_INVALID subRC_APP_DNM_NORMALIZATION_ALREADY_EXISTS subRC_DB_NDM_DATABASE_ERROR subRC_APP_NDM_ERROR subRC_SUCCESS

Table 690 Create DNM Normalization Return Codes

Table 691 lists the input fields for this method.

Field Name	Required	Description	Length	Range	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
DnmNormalization Key	Yes	This structure has the mandatory fields which define uniquely in the OSCV a DNM Normalization entry. These fields are given below			
DnmNormalization Data	Yes	This structure has all the other fields apart from the key that comprise a DNM Normalization entry. These fields are given below			
(DnmNormalizatio nKey) BgName	Yes	The name of the Business Group. Empty XML tag implies usage of BgName ‘ANY’	Char array	30 (0…30)	N/A
(DnmNormalizatio nKey) NpName	Yes	The name of the Numbering Plan. Empty XML tag implies usage of NP ‘ANY’	Char array	30 (0…30)	N/A
(DnmNormalizatio nKey) EpName	Yes	The name of the Endpoint that this entry belongs	Char array	16 (0…16)	N/A
(DnmNormalizatio nKey) InputPattern	Yes	The pattern based on which the input number will be normalized	Char array	64 (1…64)	N/A

Table 691 Create DNM Normalization Input Fields(Sheet 1 of 2)

Field Name	Required	Description	Length	Range	Default Value
(DnmNormalizationKey) InputTypeOfNumber	Yes	The input type of number	Enumerator	DNM_TON_Unknown, DNM_TON_International, DNM_TON_National, DNM_TON_Subscriber, DNM_TON_L2, DNM_TON_L1, DNM_TON_L0	DNM_TON_Unknown
(DnmNormalizationData) NormalizedTypeOfNumber	No	The normalized (output) type of number	Enumerator	DNM_TON_International, DNM_TON_L2, DNM_TON_L1, DNM_TON_L0	DNM_TON_International

Table 691 Create DNM Normalization Input Fields(Sheet 2 of 2)

Table 692 lists the output fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.

Table 692 Create DNM Normalization Output Fields

13.4.4 Modify DNM Normalization

Brief Service description: Using this web service, the craft person will be able to modify an existing entry in the display_number_normalization_t table of DB through NDM. If the optional input/output TONs are not passed in this request, the previous values would remain intact. All other fields are mandatory. Because of the fact that for a given endpoint, more than one inputPattern are allowed to be defined, the search of the existing entry to be modified is based on its index, stored in the DB. This index is returned from the query operation.

Table 693 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.01.03	subRC_APP_DSA_BLOCKED_BY_OP_MODE subRC_APP_CONGESTION_OVERLOAD subRC_APP_INSUFFICIENT_PERMISSIONS subRC_APP_COMM_OPERATOR_INVALID subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_NAME_LEN_INVALID subRC_APP_DNM_NP_INVALID subRC_APP_DNM_EP_INVALID subRC_APP_DNM_INPUT_PATTERN_INVALID subRC_APP_DNM_INPUT_TON_INVALID subRC_APP_DNM_NORMALIZED_TON_INVALID subRC_APP_DNM_OUTPUT_EXPRESSION_INVALID subRC_APP_DNM_NORMALIZATION_WAS_NOT_FOUND subRC_DB_NDM_DATABASE_ERROR subRC_APP_NDM_ERROR subRC_SUCCESS

Table 693 Modify DNM Normalization Return Codes

Table 694 lists the input fields for this method.

Field Name	Required	Description	Length	Range	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
DnmNormalization Key	Yes	This structure has the mandatory fields which define uniquely in the OSCV a DNM Normalization entry. These fields are given in the Create DNM Normalization operation.			
DnmNormalization ModKey	No	The same type as the above. This is used in order to be able to modify the key structure. InputPattern and InputTON field are modifiable			
DnmNormalization Data	Yes	This structure has all the other fields apart from the key that comprise a DNM Normalization entry. These fields are given in the Create DNM Normalization description. Note: In case the optional enumerator values (NormalizedTypeOfNumber) are not passed (the tag doesn't exist in the request), then the old values remain intact			

Table 694 Modify DNM Normalization Input Fields

Table 695 lists the output fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.

Table 695 Modify DNM Normalization Output Fields

13.4.5 Delete DNM Normalization

Brief Service description: Using this web service, the craft person will be able to delete one or more (list) entries of the display_number_normalization_t table of DB. In case of success an informative message will be returned. In case of an error, the exact entry(ies) for which the error was raised will be returned along with the error. Rollback of the entire operation is supported through a flag present in the request (Transactional, please see below)

Table 696 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.01.03	subRC_APP_DSA_BLOCKED_BY_OP_MODE subRC_APP_CONGESTION_OVERLOAD subRC_APP_INSUFFICIENT_PERMISSIONS subRC_APP_COMM_OPERATOR_INVALID subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_NAME_LEN_INVALID subRC_APP_DNM_NP_INVALID subRC_APP_DNM_EP_INVALID subRC_APP_DNM_INPUT_PATTERN_INVALID subRC_APP_DNM_NORMALIZATION_WAS_NOT_FOUND subRC_DB_NDM_DATABASE_ERROR subRC_APP_NDM_ERROR subRC_SUCCESS

Table 696 Delete DNM Normalization Return Codes

Table 697 lists the input fields for this method.

Field Name	Required	Description	Length	Range	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A

Table 697 Delete DNM Normalization Input Fields(Sheet 1 of 2)

Field Name	Required	Description	Length	Range	Default Value
Transactional	No	This flag indicates whether processing of requested list of Definitions is to be treated as a transactional block where all entries are expected to succeed or fail as a whole. When set to True, atomic operations will be followed by a Commit/Rollback statement allowing them to be treated as a single block. Potential failed atomic operations are identified and displayed as part of the result of this API when Transactional field set to False	Boolean	[true|false]	false
DeleteDnmNormalizationList	Yes	A list specifying all the entries to be deleted. Each element of the list is of type DnmNormalizationKey described in the Create DNM Normalization operation			

Table 697 Delete DNM Normalization Input Fields(Sheet 2 of 2)

Table 698 lists the output fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.
ErrorList	A structure that will be filled with the entries for which their deletion failed. If the operation was successful for all the entries, this structure will not be returned.

Table 698 Delete DNM Normalization Output Fields

13.4.6 Query DNM Normalization

Brief Service description: Using this web service, the craft person will be able to retrieve one or more (list) entries of the display_number_normalization_t table of DB through NDM. In case of success, the requested object or a series of objects (list) matching the input filtering criteria (please see below) will be returned. In case of an error, the error will be returned. This interface will support filtering on Business Group Name and/or Number Plan Name and/or Endpoint Name. Furthermore, this web service will support paging and sorting; The default sorting will be based on the weights calculated by NDM for each Normalization entry stored in DB. Sorting based on Business Group Name, Number Plan Name or Endpoint Name will also be supported.

Table 699 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.01.03	subRC_APP_DSA_BLOCKED_BY_OP_MODE subRC_APP_CONGESTION_OVERLOAD subRC_APP_INSUFFICIENT_PERMISSIONS subRC_APP_COMM_OPERATOR_INVALID subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_NAME_LEN_INVALID subRC_APP_DNM_NP_INVALID subRC_APP_DNM_EP_INVALID subRC_APP_DNM_INPUT_PATTERN_INVALID subRC_APP_DNM_NORMALIZATION_WAS_NOT_FOUND subRC_DB_NDM_DATABASE_ERROR subRC_APP_NDM_ERROR subRC_SYS_SUB_FEATURE_NO_MEMORY subRC_SUCCESS

Table 699 Query DNM Normalization Return Codes

Table 700 lists the input fields for this method.

Field Name	Required	Description	Length	Range	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1..40)	N/A
DmmQueryOptions	No	A structure containing query options for all the DNM Library's objects			
(DmmQueryOptions) Paging	No	Optional Paging structure that will be used when part of the result is required			
(DmmQueryOptions) (Paging) PageSize	No	Page size of the retrieved result set	Unsigned Integer	[1-1000]	0 (No paging)
(DmmQueryOptions) (Paging) PageNumber	No	Page number of the retrieved result set	Unsigned Integer	[1-1000]	0 (No paging)
(DmmQueryOptions) SortType	No	The sort column of the retrieved result set	Enumerator	DNM_SortType_NoSorting, DNM_SortType_Weights, DNM_SortType_BgName, DNM_SortType_NpName, DNM_SortType_EpName	DNM_SortType_Weight

Table 700 Query DNM Normalization Input Fields(Sheet 1 of 2)

Field Name	Required	Description	Length	Range	Default Value
(DmmQueryOptions) SortOrder	No	The sort order of the retrieved result set	Enumerator	Ascending Descending	Ascending
DnmNormalizationKey	No	This structure has the mandatory fields which define uniquely in the OSCV a DNM Normalization entry. These fields are given in the Create DNM Normalization operation For the query operation, all field are optional.			

Table 700 Query DNM Normalization Input Fields(Sheet 2 of 2)

Table 701 lists the output fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages.
PaginatedInfo	A common paging structure containing Current Page/Number of Pages/Page Number/Page Size/. Returned only in the case of retrieval of more than one objects.
DNMQueryNormalizationList	A list containing the actual Normalization objects stored in DB. Each item of the list is of type QueryDnmNormalizationElement which consists of a part of type DNMNormalizationKey and another of type DNMNormalizationData. Returned only on success.

Table 701 Query DNM Normalization Output Fields

13.5 DNM Conversions

This is a new table that specifies the number conversion entries that can be created to convert a public normalized number to a private normalized number or a private normalized number to a public normalized number. A special Auto-Reverse parameter allows the administrator to request an automatic creation of the reverse entry; i.e. if a public to private conversion is specified then setting the auto-reverse option will trigger the system to automatically create the private to public number conversion.

The administrator can specify numbering plan specific, business group wide and system wide entries. If the input type of number indicates a public number (International) then the output type of number must indicate a private number (L2, L1 or L0) and vice versa if the input type of number indicates a private number (L2, L1 or L0) then the output type of number must indicate a public number (International).

This table allows specifying an input pattern used for matching the number to be converted and then specifying an output modification rule and the resulting type of number of the converted output number. Just like the input pattern of the normalizations table, this input pattern allows the same wildcards and regular expressions to be used and contains information on how to split a matching digit string in fields that can be repeated in the output modification rule.

For each entry in the conversions table, the DNM library automatically creates an entry for matching an international public number and converting it to a fully qualified private number (FQPN) and another entry for matching an FQPN and converting it to an international public number if the auto-reverse option is used.

The same weighting as used for the DNM Normalization table ensures that the 498972231212 input pattern entry takes precedence over the less specific 4989722-31XXX entry which also matches for the given input number.

The following web services will be implemented from the OSCV soapServer component to accommodate this new interface's functionality.

Web Service	Description
unsp__CreateDnmConversion	Allows the creation of a new DNM Conversion entry
unsp__ModifyDnmConversion	Allows the modification of an existing DNM Conversion entry
unsp__DeleteDnmConversion	Allows the deletion of one or more (list) DNM Conversion entries
unsp__QueryDnmConversion	Allows the retrieval of one or more (list) DNM Conversion entries

Table 702 DNM Conversion Web Services

13.5.1 Create DNM Conversion

Brief Service description: Using this web service, the craft person will be able to create an entry in the display_number_conversion_t table of DB through NDM. On success, the specified entry will be created and an informative message will be returned to the client. On failure, the error will be returned.

Table 703 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.01.03	subRC_APP_DSA_BLOCKED_BY_OP_MODE subRC_APP_CONGESTION_OVERLOAD subRC_APP_INSUFFICIENT_PERMISSIONS subRC_APP_COMM_OPERATOR_INVALID subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_NAME_LEN_INVALID subRC_APP_DNM_NP_INVALID subRC_APP_DNM_INPUT_PATTERN_INVALID subRC_APP_DNM_INPUT_TON_INVALID subRC_APP_DNM_CONVERTED_TON_INVALID subRC_APP_DNM_OUTPUT_EXPRESSION_INVALID subRC_APP_DNM_CONVERSION_ALREADY_EXISTS subRC_DB_NDM_DATABASE_ERROR subRC_APP_NDM_ERROR subRC_SUCCESS

Table 703 Create DNM Conversion Return Codes

Table 704 lists the input fields for this method.

Field Name	Required	Description	Type	Range	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
DnmConversionKey	Yes	This structure has the mandatory fields which define uniquely in the OSCV a DNM Conversion entry. These fields are given below			
DnmConversionData	Yes	This structure has all the other fields apart from the key that comprise a DNM Conversion entry. These fields are given below			
(DnmConversionKey) BgName	Yes	The name of the Business Group. Empty XML tag implies usage of BgName ‘ANY’	Char array	30 (0…30)	N/A
(DnmConversionKey) NpName	Yes	The name of the Numbering Plan. Empty XML tag implies usage of NP ‘ANY’	Char array	30 (0…30)	N/A
(DnmConversionKey) InputPattern	Yes	The pattern based on which the input number will be converted	Char array	64 (1…64)	N/A
(DnmConversionKey) InputTypeOfNumber	Yes	The input type of number	Enumerator	DNM_TON_International, DNM_TON_L2, DNM_TON_L1, DNM_TON_L0	DNM_TON_International
(DnmConversionData) ConvertedTypeOfNumber	No	The converted (output) type of number	Enumerator	DNM_TON_International, DNM_TON_L2, DNM_TON_L1, DNM_TON_L0	DNM_TON_L2

Table 704 Create DNM Conversion Input Fields (Sheet 1 of 2)

Field Name	Required	Description	Type	Range	Default Value
(DnmConversionData) OutputExpression	Yes	This pattern based on the inputPattern defines the format of the converted number	Char array	64 (1…64)	N/A
(DnmConversionData) AutoReverse	No	if a public to private conversion is specified then setting the auto-reverse option will trigger the system to automatically create the private to public number conversion and vice versa	Boolean	[true|false]	false

Table 704 Create DNM Conversion Input Fields (Sheet 2 of 2)

Table 705 lists the output fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages.

Table 705 Create DNM Conversion Output Fields

13.5.2 Modify DNM Conversion

Brief Service description: Using this web service, the craft person will be able to modify an existing entry in the display_number_conversion_t table of DB through NDM On success, the specified entry will be modified and an informative message will be returned to the client. On failure, the error will be returned. Because of the fact that for a given Business Group and Numbering Plan, different inputPatterns are allowed to be defined, the search of the existing entry to be modified is based on the index of it stored in the DB. This index is returned from the query operation.

Table 706 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.01.03	subRC_APP_DSA_BLOCKED_BY_OP_MODE subRC_APP_CONGESTION_OVERLOAD subRC_APP_INSUFFICIENT_PERMISSIONS subRC_APP_COMM_OPERATOR_INVALID subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_NAME_LEN_INVALID subRC_APP_DNM_NP_INVALID subRC_APP_DNM_INPUT_PATTERN_INVALID subRC_APP_DNM_INPUT_TON_INVALID subRC_APP_DNM_CONVERTED_TON_INVALID subRC_APP_DNM_OUTPUT_EXPRESSION_INVALID subRC_APP_DNM_CONVERSION_WAS_NOT_FOUND subRC_DB_NDM_DATABASE_ERROR subRC_APP_NDM_ERROR subRC_SUCCESS

Table 706 Modify DNM Conversion Return Codes

Table 707 lists the input fields for this method.

Field Name	Required	Description	Type	Range	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
DnmConversionKey	Yes	This structure has the mandatory fields which define uniquely in the OSCV a DNM Conversion entry. These fields are given in the Create DNM Conversion operation. Only the InputPattern field is modifiable			
DnmConversionModKey	No	The same type as the above. This is used in order to be able to modify the key structure. Only the InputPattern field is modifiable			
DnmConversionData	Yes	This structure has all the other fields apart from the key that comprise a DNM Conversion entry. These fields are given in the Create DNM Conversion operation. Note: In case the two optional enumerator fields are not passed (InputTypeOfNumber, ConvertedTypeOfNumber) (the tag doesn't exist in the request), then the old values remain intact. The same goes for the field (AutoReverse)			

Table 707 Modify DNM Conversion Input Fields

Table 708 lists the output fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages.

Table 708 Modify DNM Conversion Output Fields

13.5.3 Delete DNM Conversion

Brief Service description: Using this web service, the craft person will be able to delete one or more (list) entries of the display_number_conversion_t table of DB. In case of success an informative message will be returned. In case of an error, the exact entry for which the error was raised will be returned along with the error. Rollback of the entire operation is supported in case an error is detected, through a flag present in the request (Transactional, please see below).

Table 709 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.01.03	subRC_APP_DSA_BLOCKED_BY_OP_MODE subRC_APP_CONGESTION_OVERLOAD subRC_APP_INSUFFICIENT_PERMISSIONS subRC_APP_COMM_OPERATOR_INVALID subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_NAME_LEN_INVALID subRC_APP_DNM_NP_INVALID subRC_APP_DNM_INPUT_PATTERN_INVALID subRC_APP_DNM_CONVERSION_WAS_NOT_FOUND subRC_DB_NDM_DATABASE_ERROR subRC_APP_NDM_ERROR subRC_SUCCESS

Table 709 Delete DNM Conversion Return Codes

Table 710 lists the input fields for this method.

Field Name	Required	Description	Type	Range	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
Transactional	No	This flag indicates whether processing of requested list of Definitions is to be treated as a transactional block where all entries are expected to succeed or fail as a whole. When set to True, atomic operations will be followed by a Commit/Rollback statement allowing them to be treated as a single block. Potential failed atomic operations are identified and displayed as part of the result of this API when Transactional field set to False	Boolean	[true|false]	false

Table 710 Delete DNM Conversion Input Fields (Sheet 1 of 2)

Field Name	Required	Description	Type	Range	Default Value
DeleteDnmConversionList	A list specifying all the entries to be deleted. Each element of the list is of type DnmConversionKey described in the Create DNM Conversion operation				

Table 710 Delete DNM Conversion Input Fields (Sheet 2 of 2)

Table 711 lists the output fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages.

Table 711 Delete DNM Conversion Output Fields

13.5.4 Query DNM Conversion

Brief Service description: Using this web service, the craft person will be able to retrieve one or more (list) entries of the display_number_conversion_t table of DB through NDM. In case of success, the requested object or a series of objects (list) matching the input filtering criteria (please see below) will be returned. In case of an error, the error will be returned. This interface will support filtering on Business Group Name and/or Number Plan Name. Furthermore, this web service will support paging and sorting; Default sorting will be based on the weights calculated by NDM for each Conversion entry stored in DB. Sorting based on Business Group Name or Number Plan will also be supported.

Table 712 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.01.03	subRC_APP_DSA_BLOCKED_BY_OP_MODE subRC_APP_CONGESTION_OVERLOAD subRC_APP_INSUFFICIENT_PERMISSIONS subRC_APP_COMM_OPERATOR_INVALID subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_NAME_LEN_INVALID subRC_APP_DNM_NP_INVALID subRC_APP_DNM_INPUT_PATTERN_INVALID subRC_APP_DNM_CONVERSION_WAS_NOT_FOUND subRC_DB_NDM_DATABASE_ERROR subRC_SYS_SUB_FEATURE_NO_MEMORY subRC_APP_NDM_ERROR subRC_SUCCESS

Table 712 Query DNM Conversion Return Codes

Table 713 lists the input fields for this method.

Field Name	Required	Description	Type	Range	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1..40)	N/A
DmmQueryOptions	No	A structure containing query options for all the DNM Library's objects			
(DmmQueryOptions) Paging	No	Optional Paging structure that will be used when part of the result is required			
(DmmQueryOptions) (Paging) PageSize	No	Page size of the retrieved result set	Unsigned Integer	[1-1000]	0 (No paging)
(DmmQueryOptions) (Paging) PageNumber	No	Page number of the retrieved result set	Unsigned Integer	[1-1000]	0 (No paging)
(DmmQueryOptions) SortType	No	The sort column of the retrieved result set	Enumerator	DNM_SortType_NoSorting, DNM_SortType_Weights, DNM_SortType_BgName, DNM_SortType_NpName	DNM_SortType_Weights
(DmmQueryOptions) SortOrder	No	The sort order of the retrieved result set	Enumerator	Ascending Descending	Ascending
DnmConversionKey	No	This structure has the mandatory fields which define uniquely in the OSCV a DNM Conversion entry. These fields are given in the Create DNM Conversion operation For the query operation all fields are optional.			

Table 713 Query DNM Conversion Input Fields

Table 714 lists the output fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages.
PaginatedInfo	A common paging structure containing Current Page/Number of Pages/Page Number/Page Size/. Returned only in the case of retrieval of more than one objects.
QueryDnmConversionList	A list containing the actual Conversion objects stored in DB. Each item of the list is of type QueryDnmConversionEntry which consists of a part of type DnmConversionKey and another of type DnmConversionData. Returned only in success.

Table 714 Query DNM Conversion Output Fields

13.6 DNM Local Toll

This is a new table that specifies a list of local exchange codes that are local within a specified area code and country code and also includes an indication on whether the area code (AC) needs to be included as part of the Subscriber level number or not.

The following table shows some examples of numbers presented to the local toll table and how the table treats the number:

International Number	Resulting Number	Description
International, 498972211111	International, 498972211111	The country codes don't match; therefore the number remains international
International, 15615551212	National, 5615551212	Country code 1 and area code 561 match, but 555 is **not** found in the local exchange code list. Therefore, the output number is a national number.
International, 15615581212	Subscriber, 5581212	Country code 1 and area code 561 match, and 558 is found in the local exchange code list. Therefore, the output number is a subscriber number. The area code is not part of the subscriber number because the check box Include AC is not checked.
International, 19542341111	Subscriber, 9542341111	Country code 1 and area code 954 match, and 234 is found in the local exchange code list. Therefore, the output number is a subscriber number. The area code is part of the subscriber number because the check box Include AC is checked.
International, 19542221111	National, 9542221111	Country code 1 and area code 954 match, but 222 is **not** found in the local exchange code list. Therefore, the output number is a national number.

Table 715 Local Toll Table Examples

The default rule for a local toll table is that all local exchange codes are local within the specified area code and country code and that the area code is not part of the local number. This rule can be specified using the single wildcard '*' for the list of matching local exchange codes.

No other wildcards or regular expressions are supported on the list of local exchange codes.

Local Toll tables can be specified system wide or business group specific.

The exchange codes recorded in a local toll table must all have the same length. It will be allowed to have multiple local toll exchange codes entries with the same area code for the same local toll table. E.g. in the example above it would be allowed to enter another local exchange codes table with the area code 561.

Local Toll table entries can be assigned to entries in the definitions table (see 2.2.1.2) with the following restrictions:

- A Local Toll table specified for a specific business group cannot be assigned to a system wide definition entry. The reason for this is that system wide definition entries can be seen by every administrator and administrators of other business groups may not have access to other business group resources.
- A Local Toll table specified for a specific business group cannot be assigned to a definition that is specific to another business group for the same reasons as above.

13.6.1 Create DNM LocalToll

Brief Service description: Using this web service, the craft person will be able to create an entry in the local_toll_t table and one or more entries in the local_toll_area_exchange_codes_t table of DB through NDM. On success, the specified entry will be created and an informative message will be returned to the client. In case of an error, the exact entry for which the error was raised will be returned along with the error. Rollback of the entire operation is supported in case of error through a flag present in the request (Transactional, please see below).

Table 716 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.01.03	subRC_APP_DSA_BLOCKED_BY_OP_MODE subRC_APP_CONGESTION_OVERLOAD subRC_APP_INSUFFICIENT_PERMISSIONS subRC_APP_COMM_OPERATOR_INVALID subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_NAME_LEN_INVALID subRC_APP_DNM_LOCAL_TOLL_NAME_INVALID subRC_APP_DNM_COUNTRY_CODE_INVALID subRC_APP_DNM_AREA_CODE_INVALID subRC_APP_DNM_EXCHANGE_CODE_INVALID subRC_APP_DNM_LOCAL_TOLL_ALREADY_EXISTS subRC_DB_NDM_DATABASE_ERROR subRC_APP_NDM_ERROR subRC_SUCCESS

Table 716 Create DNM LocalToll Return Codes

Table 717 lists the input fields for this method.

Field Name	Required	Description	Type	Range	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
DnmLocalTollKey	Yes	This structure has the mandatory fields which define uniquely in the OSCV a DNM LocalToll entry. These fields are given below			
(DnmLocalTollKey) BgName	Yes	The name of the Business Group. Empty XML tag implies usage of BgName 'ANY'	Char array	30 (1…30)	N/A
(DnmLocalTollKey) LocalTollName	Yes	The name of the Local Toll Entry	Char array	30 (1…30)	N/A
(DnmLocalTollData) CountryCode	No	The Country Code of the entry	Char array	16 (1…16)	N/A
(DnmLocalTollData) HomeAreaCode	No	Area code of the Entry	Char array	16 (1…16)	N/A
(DnmLocalTollData) HomeExchangeCode	No	Exchange code of the entry	Char array	10(1… 10)	N/A
(DnmLocalTollData) TTInternational	No	International Traffic Type	String		
(DnmLocalTollData) TTLongDistance	NO	Long Distance Traffic Type	String		
(DnmLocalTollData) TTExtendedCallingArea	NO	Extended Calling Area Traffic Type	string		
(DnmLocalTollAreaCodeList) DialPattern	No	Dialing Pattern determination	Enum	DNM_DP_Unknown, DNM_DP_National, DNM_DP_SubscriberWithAreaCode, DNM_DP_Subscriber	DNM_DP_Unknown
DnmLocalTollAreaCodeList	Yes	A list consisting of one or more items of type DnmLocalTollAreaCodeElement specifying the Area Code with its Exchange Codes defined for this entry			
(DnmLocalTollAreaCodeList) AreaCode	Yes	The Area Code of this entry	Char array	16 (1…16)	
(DnmLocalTollAreaCodeList) FirstLocalExchangeCode	Yes	The first Local Exchange Code element of a possible list that would follow. If this is the special character '*' the list should be empty. This entry is unique for a specific area code inside a Local Toll Entry	Char array	6 (1…6)	

Table 717 Create DNM LocalToll Input Fields (Sheet 1 of 2)

Field Name	Required	Description	Type	Range	Default Value
(DnmLocalTollAreaCodeList) IncludeAreaCode	No	The area code field needs to be included as part of the Subscriber level number (true) or not (false)	Boolean	[true|false]	
(DnmLocalTollAreaCodeList) LocalExchangeCodes	No	A list of comma or space separated digit strings	Char array	2500 (0…2500)	
(DnmLocalTollAreaCodeList) DialPattern	No	Dialing Pattern determination	Enum	DNM_DP_Unknown, DNM_DP_National, DNM_DP_SubscriberWithAreaCode, DNM_DP_Subscriber	

Table 717 Create DNM LocalToll Input Fields (Sheet 2 of 2)

Table 718 lists the output fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages.

Table 718 Create DNM LocalToll Output Fields

13.6.2 Modify DNM LocalToll

Brief Service description: Using this web service, the craft person will be able to modify an existing entry in the local_toll_t table of DB through NDM. On success, the specified entry will be modified and an informative message will be returned to the client. On failure, the error will be returned.The LocalExchangeCodes list will actually be deleted and created will the new entries specified in the modification request. If the list containing the LocalExchangeCodes.

Table 719 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.01.03	subRC_APP_DSA_BLOCKED_BY_OP_MODE subRC_APP_CONGESTION_OVERLOAD subRC_APP_INSUFFICIENT_PERMISSIONS subRC_APP_COMM_OPERATOR_INVALID subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_NAME_LEN_INVALID subRC_APP_DNM_LOCAL_TOLL_NAME_INVALID subRC_APP_DNM_COUNTRY_CODE_INVALID subRC_APP_DNM_AREA_CODE_INVALID subRC_APP_DNM_EXCHANGE_CODE_INVALID subRC_APP_DNM_LOCAL_TOLL_WAS_NOT_FOUND subRC_DB_NDM_DATABASE_ERROR subRC_APP_NDM_ERROR subRC_SUCCESS

Table 719 Modify DNM LocalToll Return Codes

Table 720 lists the input fields for this method.

Field Name	Required	Description	Type	Range	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
DnmLocalTollKey	Yes	This structure has the mandatory fields which define uniquely in the OSCV a DNM Local Toll entry. These fields are given in the Create DNM Local Toll operation.			
DnmLocalTollModKey	No	The same type as the above. Contains the new possible key for this object. All but the BgName field are modifiable			
DnmLocalTollAreaCodeList	Yes	A list consisting of one or more items specifying the new Area Code objects along their Local Exchange Codes defined for this entry. This type is given in the Create DNM Local Toll operation.			
(DnmLocalTollData) TTInternational	No	International Traffic Type	String		
(DnmLocalTollData) TTLongDistance	NO	Long Distance Traffic Type	String		
(DnmLocalTollData) TTExtendedCallingArea	NO	Extended Calling Area Traffic Type	string		

Table 720 Modify DNM LocalToll Input Fields

Table 721 lists the output fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages.

Table 721 Modify DNM LocalToll Output Fields

13.6.3 Delete DNM LocalToll

Brief Service description: Using this web service, the craft person will be able to delete one or more (list) entries of the local_toll_t table of DB. All the etries from local_toll_area_echange_codes_t table having references to the one that is deleted from local_toll_t table will be deleted also. In case of success an informative message will be returned. In case of an error, the exact entry for which the error was raised will be returned along with the error. Rollback of the entire operation is supported in case of an error through a flag present in the request (Transactional, please see below).

Table 722 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.01.03	subRC_APP_DSA_BLOCKED_BY_OP_MODE subRC_APP_CONGESTION_OVERLOAD subRC_APP_INSUFFICIENT_PERMISSIONS subRC_APP_COMM_OPERATOR_INVALID subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_NAME_LEN_INVALID subRC_APP_DNM_LOCAL_TOLL_NAME_INVALID subRC_APP_DNM_COUNTRY_CODE_INVALID subRC_APP_DNM_LOCAL_TOLL_WAS_NOT_FOUND subRC_DB_NDM_DATABASE_ERROR subRC_APP_NDM_ERROR subRC_SUCCESS

Table 722 Delete DNM LocalToll Return Codes

Table 723 lists the input fields for this method.

Field Name	Required	Description	Type	Range	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A

Table 723 Delete DNM LocalToll Input Fields (Sheet 1 of 2)

Field Name	Required	Description	Type	Range	Default Value
Transactional	No	This flag indicates whether processing of requested list of Definitions is to be treated as a transactional block where all entries are expected to succeed or fail as a whole. When set to True, atomic operations will be followed by a Commit/Rollback statement allowing them to be treated as a single block. Potential failed atomic operations are identified and displayed as part of the result of this API when Transactional field set to False	Boolean	[true|false]	false
DeleteDnmLocalTollList	A list specifying all the entries to be deleted. Each element of the list is of type DnmLocalTollKey which is given in the Create DNM Local Toll operation				

Table 723 Delete DNM LocalToll Input Fields (Sheet 2 of 2)

Table 724 lists the output fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages.
ErrorList	A structure that will be filled with the entries for which their deletion failed. If the operation was successful for all the entries, this structure will not be returned.

Table 724 Delete DNM LocalToll Output Fields

13.6.4 Query DNM LocalToll

Brief Service description: Using this web service, the craft person will be able to retrieve one or more (list) entries of the local_toll_t table of DB through NDM. In case of success, the requested object or a series of objects (list) matching the input filtering criteria (please see below) will be returned. In case of an error, the error will be returned. This interface will support filtering on Business Group Name. Furthermore, this web service will support paging and sorting on Business Group Name.

Table 725 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.01.03	subRC_APP_DSA_BLOCKED_BY_OP_MODE subRC_APP_CONGESTION_OVERLOAD subRC_APP_INSUFFICIENT_PERMISSIONS subRC_APP_COMM_OPERATOR_INVALID subRC_DB_BG_LOOKUP_FAILED subRC_APP_BG_DOES_NOT_EXIST subRC_APP_BG_NAME_LEN_INVALID subRC_APP_DNM_LOCAL_TOLL_NAME_INVALID subRC_APP_DNM_COUNTRY_CODE_INVALID subRC_APP_DNM_LOCAL_TOLL_WAS_NOT_FOUND subRC_DB_NDM_DATABASE_ERROR subRC_APP_NDM_ERROR subRC_SUCCESS

Table 725 Query DNM Local Toll Return Codes

Table 726 lists the input fields for this method.

Field Name	Required	Description	Type	Range	Default Value
OperatorId	Yes	I/F operator	Char array		40 (1…40)
DmmQueryOptions	No	A structure containing query options for all the DNM Library's objects			
(DmmQueryOptions) Paging	No	Optional Paging structure that will be used when part of the result is required			
(DmmQueryOptions) (Paging) PageSize	No	Page size of the retrieved result set	Unsigned Integer	[1-1000]	0 (No paging)
(DmmQueryOptions) (Paging) PageNumber	No	Page number of the retrieved result set	Unsigned Integer	[1-1000]	0 (No paging)
(DmmQueryOptions) SortType	No	The sort column of the retrieved result set	Enumerator	DNM_SortType_NoSorting, DNM_SortType_BgName	DNM_SortType_NoSorting
(DmmQueryOptions) SortOrder	No	The sort order of the retrieved result set	Enumerator	Ascending Descending	Ascending
DnmLocalTollKey	No	This structure has the mandatory fields which define uniquely in the OSCV a DNM Local Toll entry. These fields are given in the Create DNM Local Toll operation For the query operation all fields are optional			

Table 726 Query DNM Local Toll Input Fields (Sheet 1 of 2)

Field Name	Required	Description	Type	Range	Default Value
(DnmLocalTollData) TTInternational	No	International Traffic Type	String		
(DnmLocalTollData) TTLongDistance	NO	Long Distance Traffic Type	String		
(DnmLocalTollData) TTExtendedCallingArea	NO	Extended Calling Area Traffic Type	string		

Table 726 Query DNM Local Toll Input Fields (Sheet 2 of 2)

Table 727 lists the output fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages.

Table 727 Query DNM Local Toll Output Fields

14 Feature Profiles

Feature Profiles contain a set of features grouped together that can be assigned to subscribers for inheriting the features. It allows the scale of inheriting features to be more flexible and not controlled by the Business Group. It also adds the Modify subscribed status to update individual feature data. See Section 4.2, “Feature Profiles”, on page 34.

14.1 Create Feature Profile

This method creates a feature profile.

Once created, the Feature Profile Id, Feature Profile Type, and Feature Profile Connection Type cannot be changed.

Table 728 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01.12 10.00.03.01		subRC_SUCCESS subRC_APP_FP_NAME_LEN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_FP_ALREADY_EXIST subRC_APP_FP_BG_NAME_TYPE_CONFLICT subRC_DB_FP_INSERTION_FAILED
Comments:	10.00.03.01 – added the Feature Profile Connection Type.	

Table 728 CreateFeatureProfile Return Codes

Table 729 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
FeatureProfile ID	Identifies the Feature Profile; comprises the Feature Profile Name and the BGName.			
(FeatureProfile ID) Feature Profile Name	The name of the feature profile	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		required
(FeatureProfile ID) BG Name	The name of the business group with which the feature profile is associated.	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		“”

Table 729 CreateFeatureProfile Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
FeatureProfileType	Governs the set of features and types of subscribers allowed to assign to the feature profile.	Enum	FeatureProfileResidential, FeatureProfileCentrex	FeatureProfileResidential
FeatureProfileConnection Type	Determines whether the profile is applicable for which connection type. Note that normally a connection type other than ConnectionTypeNA should be provided. ConnectionTypeNA is not recommended to be used in a live environment since the mixture of subscribers with different connection types in the same Feature Profile may result in inconsistencies if features are assigned that only apply to one or the other. ConnectionTypeMGCP is not for OpenScape Voice.	Enum	ConnectionTypeNA, ConnectionTypeSIP, ConnectionTypeMGCP	Connection TypeNA – applies to all connection types.
ListofFeatures	List of features to assign to the Feature Profile	Structure	List of valid features	Null

Table 729 CreateFeatureProfile Input Fields (Sheet 2 of 2)

14.2 Update Feature Profile

This method modifies the features associated with a feature profile.

Once created, the Feature Profile Id, Feature Profile Name, Feature Profile Type, and Feature Profile Connection Type cannot be changed. Only the features for the Feature Profile are to be updated.

Table 730 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01.12		subRC_SUCCESS subRC_APP_FP_NAME_LEN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID

Table 730 UpdateFeatureProfile Return Codes

Table 731 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required

Table 731 UpdateFeatureProfile Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
FeatureProfile ID	Identifies the Feature Profile; comprises the Feature Profile Name and the BGName.			
(FeatureProfile ID) NewFeature Profile Name	The new name of the feature profile	Please refer to Section 5.4,	optional	(FeatureProfile ID) NewFeature Profile Name
(FeatureProfile ID) Feature Profile Name	The name of the feature profile	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		required
(FeatureProfile ID) BG Name	The name of the business group with which the feature profile is associated.	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		“”
ListofFeatures	List of features to assign to the Feature Profile	Structure	List of valid features	Null.
Comments:	All the subscribers associated with the Feature Profile will be updated for the inheritance if the Feature Profile is updated successfully.			

Table 731 UpdateFeatureProfile Input Fields (Sheet 2 of 2)

14.3 Delete Feature Profile

This method deletes the specified feature profile. A feature profile may not be deleted if it is referenced by a subscriber. The GetSubscriberList may be used to obtain information about subscribers that reference a Feature Profile

Table 732 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01.12		subRC_SUCCESS subRC_APP_FP_NAME_LEN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_FP_DOES_NOT_EXIST subRC_DB_FP_DELETE_FAILURE subRC_APP_FP_DELETION_FAILED subRC_APP_FEAT_IN_PROFILE_NAME_DOES_NOT_EXIST subRC_DB_FEAT_IN_PROFILE_NAME_LOOKUP_FAILED subRC_DB_FP_DELETE_FAILURE

Table 732 DeleteFeatureProfile Return Codes

Table 733 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
FeatureProfile ID	Identifies the Feature Profile; comprises the Feature Profile Name and the BGName.			
(FeatureProfile ID) Feature Profile Name	The name of the feature profile	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		required
(FeatureProfile ID) BG Name	The name of the business group with which the feature profile is associated.	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		“”
Comments:	The feature profile cannot be deleted if there is any reference to it.			

Table 733 DeleteFeatureProfile Input Fields

14.4 Get Feature Profile List

This method retrieves a list of all feature profiles on the OpenScape Voice system.

Table 734 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01.12		subRC_SUCCESS subRC_APP_BG_NAME_LEN_INVALID subRC_SYS_FP_ALLOC_ERR subRC_DB_FP_LIST_SWITCH_WIDE_FAILED subRC_DB_FP_LIST_LOOKUP_FAILED subRC_SYS_FP_ALLOC_ERR

Table 734 GetFeatureProfileList Return Codes

Table 735 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
Paging	Paging structure for Get List requests. For further details, see Section 5.5, “Paging”, on page 72.			N/A
BGName	Business Group Name. Refer to Section 5.4, “Primitive Data Definitions”, on page 66 for further details.			“”
FeatureProfileName	Feature profile name. Refer to Section 5.4, “Primitive Data Definitions”, on page 66 for further details.			“”

Table 735 GetFeatureProfileList Input Fields (Sheet 1 of 2)

Field Name	Description			Length	Range/Values	Default Value
CombinedList	Indicator to combine the switchwide and BG profile for retrieving the profile list.			Boolean	True, False	False
	BGName	CombinedList	RetrievalList			
	NIL	True	Switchwide + all BGs profile list.			
	NIL	False	Switchwide List			
	Bg Name	True	Switchwide + BG-specific profile list			
	Bg Name	False	BG-specific profile list			
FProfileListSortType	Defines the FP attribute to be used for sorting results (currently only FPName is supported)			enum	FPSortTypeFPName_	FPSortTypeFPName_
SortOrder	Defines the order by which results will be returned			enum	SortOrderAscending, SortOrderDescending	SortOrderAscending

Table 735 GetFeatureProfileList Input Fields (Sheet 2 of 2)

14.5 Get Feature Profile

This method retrieves the details of one feature profile. The GetSubscriberList may be used to obtain information about subscribers that reference a Feature Profile.

Table 736 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01.12		subRC_SUCCESS subRC_APP_FP_NAME_LEN_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_SYS_FP_AUTH_CODE_ALLOC_ERR subRC_DB_FP_LOOKUP_FAILED

Table 736 GetFeatureProfile Return Codes

Table 737 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
NewFeatureProfileID	Identifies the Feature Profile; comprises the Feature Profile Name and the BGName.			

Table 737 GetFeatureProfile Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
(FeatureProfile ID) Feature Profile Name	The name of the feature profile	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		required
(FeatureProfile ID) BG Name	The name of the business group with which the feature profile is associated.	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		“““

Table 737 GetFeatureProfile Input Fields (Sheet 2 of 2)

15 Hunt Groups

The OpenScape Voice system supports Hunt Groups (MLHG). An Hunt Group is a collection of phones that have a common pilot directory number (Pilot DN) and specific processing. An Hunt Group is identified by its Pilot DN.

A subscriber or BGL can be a member of multiple Hunt Groups. In this case, one Hunt Group is designated as the primary Hunt Group. If subscriber S is a member of Hunt Group A and is also the Pilot DN of Hunt Group B, hunting through Hunt Group B takes place only if subscriber S is called directly, not if the call is presented to subscriber S as a result of hunting after an incoming call to Hunt Group A's Pilot DN. When a subscriber or BGL is a member of multiple Hunt Groups and its primary Hunt Group is removed, the member’s primary Hunt Group designation is updated to refer to one of the still existing Hunt Groups (the next one in chronological order in which the member was added to the Hunt Groups).

Hunt Groups can consist of residential phones or BG phones, but not a combination within a single Hunt Group.

The DN to be used as a Hunt Group Pilot DN or a Hunt Group terminal DN must be created as a subscriber before being converted into a Hunt Group Pilot or terminal. The subscriber can have a connection (phone). If the Hunt Group Pilot subscriber has a phone, then that phone is the master phone for the Hunt Group and controls the Hunt Group services (CFB, CFDA, and so on). If the subscriber does not have any connection info, then the Hunt Group is just a container with its service features.

Service features can be applied to the Hunt Group Pilot DN or an Hunt Group Member (DN). When a caller calls a subscriber that is an Hunt Group Pilot DN, the services of the Pilot DN (with some exceptions) are executed — not the services of the member or terminal to which the call actually hunts and terminates to. On the other hand, when a caller directly calls a subscriber that is a member of one or more Hunt Groups, the subscriber's services are executed and not those of any Hunt Group of which the subscriber may be a member.

> An exception to the above statement is Keyset Operation. A subscriber can be a keyset phone and also be a member of a Hunt Group. In this case, it is desirable that when an incoming call reaches a keyset Hunt Group member via hunting, the call should be presented to all devices where the subscriber has a line appearance. To accomplish this, it is necessary for the Hunt Group service to start the keyset service on each individual Hunt Group member, if it is a keyset subscriber, and also to stop the service if the No Advance timer expires and hunting must continue.

Outgoing calls, which some exceptions, invoke the subscriber's features, and not those of the Hunt Group.

The OpenScape Voice system can support 25K Hunt Groups and 150K terminals system-wide. Each Hunt Group can have up to 2000 terminals. The maximum queue size for an Hunt Group is 511.

15.1 Terminals

MemberServiceId – represents a phone within a Hunt Group. It must be an E164 DN.

The MLHG contains a list of phones in a specified order, as well as the following attributes:

Position – Terminals are in an ordered list. When inserting and/or moving a terminal in the list, 1 indicates that the terminal should be inserted in the first position (at the head of the list) 0, negative, a number greater than the list size, or no position specified, indicates that the terminal is to be inserted at the end of the list. Any other number indicates the position to insert the terminal into. For Master Hunt Groups, a terminal is not allowed to be assigned to position 1. Also, the Pilot of the Master Hunt group is only allowed to be assigned to position 1.

MlhgMakeBusyStatus – indicates whether the subscriber has marked the phone as busy and not available for new incoming calls. If not defined, defaults to NotBusy.

MlhgStopHuntStatus – indicates whether the subscriber has marked the phone as StopHunt and that new incoming calls should not keep searching for an open phone beyond this one. If not defined, defaults to NotStopHunt.

StopHuntSubControlled – indicates whether the terminal (phone) is capable of controlling its StopHunt Status. This information is common to all Hunt Groups that the terminal is part of. Thus, changing this setting for the terminal in one Hunt Group will change this capability of the terminal in all Hunt Groups that the terminal is part of. If not specified when adding the terminal to a Hunt Group, it defaults to false and updates any existing capability to false.

MakeBusySubControlled – indicates whether the terminal (phone) is capable of controlling its MakeBusy Status. This information is common to all Hunt Groups that the terminal is part of. Thus, changing this setting for the terminal in one Hunt Group changes this capability in all Hunt Groups that the terminal is part of. If not specified when adding the terminal to a Hunt Group, it defaults to false and updates any existing capability to false.

15.2 Hunt Group Phone Services

Some services are not supported for provisioning to Hunt Group Pilot DNs and/or Hunt Group members. Table 738 lists these services, where **X** indicates that the feature IS NOT supported.

Service Feature	MLHG Pilot	MLHG Member
CW — Call Waiting	X	
CCW — Cancel Call Waiting	X	
CIDCW — Calling Identity Delivery on Call Waiting	X	
CSTA — ComAssistant	X (but see below)	
SRS — Simultaneous Ringing	X	
RFA — Remote Feature Access (Supports configuration of SRS)	X	
SERRNG — Serial Ringing	X	
RCF — Remote Call Forward	X	X
HotDesk — Hot Desking	X	X

Table 738 Services that Cannot be Provisioned for MLHG Pilot DNs and MLHG Members

If an attempt is made to assign any of the features listed above to a Hunt Group Pilot DN (or, in the case of RCF or HotDesk, to a Hunt Group member), the request is rejected. Likewise, an attempt to create a Hunt Group using an existing subscriber or BGL as the Pilot DN will fail if the existing DN already has one or more of the above features subscribed as AddReplace or Inherited. If the DN is denied the feature, the Hunt Group can be created.

The Speed Calling services (SPCALL and BgSPCALL) are allowed for Hunt Group Pilot DNs. The user should bear in mind, however, that these services can be used only if the Hunt Group Pilot DN is also a Hunt Group Member, or terminal. So, while a Speed Calling list can be assigned to a nonterminal Pilot DN, is cannot be used by other Hunt Group Members.

CT (Call Transfer) service is allowed for both Hunt Group members and Pilot DNs, but it must be assigned to both the Hunt Group member and the Pilot DN in order to function properly for the Hunt Group member.

For CSTA (ComAssistant), a special situation exists. It is possible to assign the CSTA service to a Hunt Group Pilot DN only if it is not also a Hunt Group Member, or terminal. When a Hunt Group is created with a Pilot DN that has connection info, it is automatically also added as a terminal. Therefore, in order to assign the CSTA service to a Hunt Group Pilot DN, the DN must be a profile-only subscriber (having no connection info).

The SOAP Server provides the capability for the operator to create a Hunt Group using and existing subscriber/BGL as the Pilot DN. The interface also allows the operator to delete and update Hunt Groups, as well as to add an existing subscriber/BGL as a member (terminal) of a Hunt Group and update the attributes for each member.

15.2.1 Create Mlhg

This method creates a Hunt Group.

Table 739 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.04.07 10.00.01.18[1] 10.00.01.19[2]		subRC_SUCCESS subRC_APP_MLHG_PILOT_DN_LEN_INVALID subRC_APP_MLHG_PILOT_DN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_MLHG_EXTERNAL_DN_LEN_INVALID subRC_APP_MLHG_OVERFLOW_DN_LEN_INVALID subRC_APP_MLHG_MAX_QUEUE_SIZE_OUT_OF_RANGE subRC_APP_MLHG_CONF_QUEUE_SIZE_OUT_OF_RANGE subRC_APP_MLHG_MAX_CONF_QUEUE_SIZE_CONFLICT subRC_APP_MLHG_NO_ANS_ADV_TIME_INVALID subRC_APP_MLHG_AUTO_MAKE_BUSY_NO_ANS_ADV_TIME_CONFLIC T subRC_APP_MLHG_INTERCEPT_LEN_INVALID subRC_APP_MLHG_NIGHT_SVC_DN_LEN_INVALID subRC_APP_MLHG_MAX_TIME_QUEUE_OUT_OF_RANGE subRC_APP_MLHG_NIGHT_SVC_DN_MANUAL_CONFLICT subRC_APP_MLHG_NIGHT_SVC_AUTO_STATUS_INVALID subRC_APP_MLHG_POST_CALL_TIMER_INVALID subRC_APP_MLHG_PILOT_DN_DOES_NOT_EXIST subRC_DB_MLHG_PILOT_DN_LOOKUP_FAILED subRC_APP_MLHG_PILOT_DN_H323_CONFLICT subRC_APP_MLHG_PILOT_DN_TERM_CONFLICT subRC_APP_MLHG_PILOT_DN_EXIST subRC_APP_MLHG_PILOT_DN_PARENT_TEEN_CONFLICT subRC_APP_MLHG_PILOT_DN_FEATURE_CONFLICT subRC_APP_MLHG_PILOT_DN_CONNECTION_CSTA_CONFLICT subRC_APP_MLHG_OVERFLOW_DN_INVALID subRC_APP_MLHG_NIGHT_SVC_DN_INVALID subRC_APP_MLHG_PILOT_DN_BG_ID_CONFLICT subRC_DB_BG_LOOKUP_FAILED subRC_APP_MLHG_MANUAL_HUNT_CSTA_CONFLICT subRC_DB_COMM_INTERNAL_RECOVERY_ERROR subRC_DB_MLHG_INSERTION_FAILED subRC_DB_MLHG_OFFICE_CODE_LOOKUP_FAILED subRC_DB_MLHG_HOME_DN_ASSIGN_FAILED subRC_DB_MLHG_INSERTION_FAILED
V5.00.20.01		subRC_APP_MLHG_PILOT_DN_OSMO_CONFLICT
[1]Added attributes introduced for FRN0124 and FRN0944. [2]Added TMStatus, introduced for FRN0617.		

Table 739 Create Mlhg Return Codes

Table 740 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
BgName	Business Group Name. Mandatory if the PilotServiceId is a BGL.	See definition of BGName in Section 5.4, "Primitive Data Definitions", on page 66.		
OperatorId	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.			
PilotServiceId	The DN of the subscriber/BGL that represents the Pilot DN of the MLHG. That is, the number that when dialed will cause hunting through the MLHG members to take place.	See definition of ServiceId in Section 5.4, "Primitive Data Definitions", on page 66.		N/A
BOName	Branch Office Name to be associated with MLHG V7.00.01.09	char	0…30	Must coincide with BOName of the Pilot DN.
HuntType	The type of hunting algorithm used by the MLHG to determine which MLHG member should be presented with incoming calls to the Pilot DN. Notes: *Linear*: The default hunting method, in which each MLHG terminal is checked in sequence for one pass only. *CircularWithHunt*: more properly referred to as *CircularWithMemory*. *UCD*: Uniform Call Distribution. Indicates that an ordered list of idle terminals is maintained and calls are presented to those that have been idle the longest. *Manual*: Indicates that calls are not presented automatically; rather, agents must view calls via the CallManager application and manually select which calls to process. *Parallel:* All available members of the Hunt Group are alerted simultaneously whenever a new call arrives at the Hunt Group.	enum	Linear CircularWithHunt UCD Manual Parallel	Linear

Table 740 Create Mlhg Input Fields (Sheet 1 of 4)

Field Name	Description	Length	Range	Default Value
OverFlowDN	Specifies the destination of unanswered incoming calls when the queue is full. This DN can be an E164DN or a private extension if this MLHG is part of a BG. If not specified, received calls that cannot be queued receive busy treatment. Optional.		See definition of ServiceId in Section 5.4, “Primitive Data Definitions”, on page 66.	“”
ConfQueue Size	Indicates the number of calls that may be queued waiting for an available phone (terminal). If not specified, the default value is MaxQueueSize.	integer	0..MaxQueueSize	MaxQueueSize
MaxQueue Size	Indicates the maximum number of calls that may be queued waiting for an available phone (terminal). This parameter can be configured by the telco and sets a hard limit of the queue size.	integer	0..511	0
CallingParty Source	Indicates which calling number should be used on outgoing calls from a device that is a terminal in the MLHG.	enum	PilotDN UseLineAndPilotDN Note: PilotDN option is currently not supported. Hunt Group will function according to UseLineAndPilotDN option, regardless of the chosen setting.	UseLineAnd PilotDN
NoAnsAdv Time	Indicates the time for which a terminal will ring before the hunting algorithm moves on to continue searching for the next available terminal.	integer	5..3600 (seconds)	29
Intercept Annoncement Id	Indicates the configured intercept announcement that is to be played when an incoming call is queued because no terminals are available. Optional	0..15	Character string – must represent a valid intercept.[1]	MLHG_Default
NoInterceptAnnou ncement	When set to True, no announcement is provided from the OSV platform media server toward the queued party. Instead the queued party will hear local ringback provided by the network or subscriber device.	Boolean	True, False	False

Table 740 Create Mlhg Input Fields (Sheet 2 of 4)

Field Name	Description	Length	Range	Default Value
NightService DN	The directory number to which calls are diverted when the Night Service feature is active. If provided, the SOAP Server validates the DN via a translation routine to ensure that it is routable at the time of provisioning. If it is not routable, the request is rejected. Optional.		See definition of ServiceId in Section 5.4, "Primitive Data Definitions", on page 66.	""
MaxTimeIn Queue	Monitors the time in queue for calls to the MLHG. When the actual time exceeds the provisioned value and other conditions are met, the queued call is removed and routed to the OverflowDN, if any. If there is no OverflowDN, the call is routed to the Night Service DN, if any. If there is no Night Service DN, the call receives busy tone. If not provided, the default is 0, which is interpreted by call processing to mean that the value is not provisioned and a queued call will remain in the queue indefinitely.	integer	0..43200	0
NightService ManualStatus	Enables the operator to manually activate Night Service.	Boolean	True, False	False
PostCallTimer	Number of seconds to pause after each call before presenting next call to same agent. If not provided, the default value of 0 is used, which indicates that there shall be no post call pause. Note that this value is ignored if the HuntType is Manual. Optional.	integer	0..43200	0
TMStatus	Enables the operator to enable or disable Traffic Measurements for this MLHG.	Boolean	True, False	False
InQueueInterruptIntercept	Specifies the intercept to be played to callers in the MLHG's queue periodically after the number of seconds specified in the InQueueInterruptTime. NOTE: This attribute is not supported for Version 2.2 WSDL versions!	0..15	Character string – must represent a valid intercept.[1]	MLHG_Q_Pos

Table 740 Create Mlhg Input Fields (Sheet 3 of 4)

Field Name	Description	Length	Range	Default Value
IsMlhgReroutingAllowed	This attribute can be provisioned to allow or deny subscriber rerouting. If the MLHG is provisioned to allow subscriber rerouting, the Subscriber rerouting will only take place for MLHG members that are provisioned for subscriber rerouting	Boolean	True, False	False
SendPilotDisplay	Controls whether the OpenScape Voice sends the Pilot's display information to the Hunt Group members for hunted calls	Boolean	True, False	False
isMlhgEnhancedRerouteAllowed	Controls whether Enhanced Subscriber Rerouting is allowed in case a distributed call to a Hunt Group member is rejected due to Call Admission Control. The default value for this parameter shall be disabled.	Boolean	True, False	False
OverflowDnNoRerouting	Define san Overflow destination in case a distributed call to the Hunt Group member is rejected due to Call Admission Control or WAN Failure AND Subscriber Rerouting (Enhanced or Basic) either fails or is not allowed.	String		
AllowCallsToOverflow	Allow calls to overflow (Unconditional). Applies for Profile Only and SIP Master Pilot Hunt Group DNs.	Boolean	True, False	
AllowUnansweredCallsToRequeue	Applies for Profile Only and SIP Master Pilot Hunt Group DNs.	Boolean	True, False	
[1]The list of valid existing intercepts is available using the *GetInterceptList* SOAP request (see Section 7.8, "Get Intercept List").				

Table 740 Create Mlhg Input Fields (Sheet 4 of 4)

15.2.2 Modify Mlhg Info

This method adds, removes, and updates Hunt Group info.

Table 741 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.04.07 10.00.01.18[1] 10.00.01.19[2]		subRC_APP_COMM_OPERATOR_INVALID subRC_DB_MLHG_LOOKUP_FAILED subRC_DB_MLHG_PILOT_DN_LOOKUP_FAILED subRC_APP_MLHG_MANUAL_HUNT_CSTA_CONFLICT subRC_APP_MLHG_HUNT_TYPE_INVALID
[1]Added attributes introduced for FRN0124 and FRN0944. [2]Added TMStatus, introduced for FRN0617.		

Table 741 Modify Mlhg Info Return Codes

Table 742 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
BgName	Business Group Name. Mandatory if the PilotServiceId is a BGL.	See definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
BOName	Branch Office Name to be associated with MLHG V7.00.01.09	char	0…30	Must coincide with BOName of the Pilot DN.
PilotServiceId	The DN of the subscriber/BGL that represents the Pilot DN of the MLHG. That is, the number that when dialed will cause hunting through the MLHG members to take place.	See definition of ServiceId in Section 5.4, “Primitive Data Definitions”, on page 66.		N/A

Table 742 Modify Mlhg Info Input Fields (Sheet 1 of 5)

Field Name	Description	Length	Range/Values	Default Value
HuntType	The type of hunting algorithm used by the MLHG to determine which MLHG member should be presented with incoming calls to the Pilot DN. Notes: *Linear*: The default hunting method, in which each MLHG terminal is checked in sequence for one pass only. *CircularWithHunt*: is more properly referred to as *CircularWithMemory*. *UCD*: Uniform Call Distribution. This indicates that an ordered list of idle terminals is maintained and calls are presented to those that have been idle the longest. *Manual*: Indicates that calls are not presented automatically; instead, agents must view calls via the CallManager application and manually select which calls to process. *Parallel:* All available members of the Hunt Group are alerted simultaneously whenever a new call arrives at the Hunt Group	enum	Linear CircularWithHunt UCD Manual Parallel	Linear
OverFlowDN	Specifies the destination of unanswered incoming calls when the queue is full. This DN can be an E164DN or a private extension if this MLHG is part of a BG. If not specified, received calls that cannot be queued receive busy treatment. Optional.	See definition of ServiceId in Section 5.4, “Primitive Data Definitions”, on page 66.		N/A
ConfQueue Size	Indicates the number of calls that may be queued waiting for an available phone (terminal). If not specified, the default value is MaxQueueSize.	integer	0..MaxQueueSize	N/A
MaxQueue Size	Indicates the maximum number of calls that may be queued waiting for an available phone (terminal). This parameter can be configured by the telco and sets a hard limit of the queue size.	integer	0..511	N/A

Table 742 Modify Mlhg Info Input Fields (Sheet 2 of 5)

Field Name	Description	Length	Range/Values	Default Value
CallingParty Source	Indicates which calling number should be used on outgoing calls from a device that is a terminal in the MLHG.	enum	PilotDN UseLineAndPilotDN Note: PilotDN option is currently not supported. Hunt Group will function according to UseLineAndPilotDN option, regardless of the chosen setting.	N/A
NoAnsAdv Time	Indicates the time for which a terminal will ring before the hunting algorithm moves on to continue searching for the next available terminal.	integer	5..3600 (seconds)	N/A
Intercept Annoncement Id	Indicates the configured intercept announcement that is to be played when an incoming call is queued because no terminals are available. Optional.	15	Character string – must represent a valid intercept.[1]	N/A
NightService DN	The directory number to which calls are diverted when the Night Service feature is active. If provided, the SOAP Server validates the DN via a translation routine to ensure that it is routable at the time of provisioning. If it is not routable, the request is rejected. Optional.	See definition of ServiceId in Section 5.4, “Primitive Data Definitions”, on page 66.		N/A
AutoMakeBusy	Indicates whether DNs in MLHG are marked busy on auto advance timer expiry.	Boolean	True, False	N/A
MaxTimeIn Queue	Used to monitor the time in queue for calls to the MLHG. When the actual time exceeds the provisioned value and other conditions are met, the queued call is removed and routed to the OverflowDN, if any. If there is no OverflowDN, the call is routed to the Night Service DN, if any. If there is no Night Service DN, the call receives busy tone. If not provided, the default is 0, which is interpreted by call processing to mean that the value is not provisioned and a queued call will remain in the queue indefinitely.	integer	0..600	N/A
NightService ManualStatus	Enables the operator to manually activate Night Service.	Boolean	True, False	N/A

Table 742 Modify Mlhg Info Input Fields (Sheet 3 of 5)

Field Name	Description	Length	Range/Values	Default Value
PostCallTimer	Number of seconds to pause after each call before presenting next call to same agent. If not provided, the default value of 0 is used, which indicates that there shall be no post call pause. Note that this value is ignored if the HuntType is Manual. Optional.	integer	0..120	N/A
TMStatus	Enables the operator to enable or disable Traffic Measurements for this MLHG.	Boolean	True, False	N/A
InQueueInterruptTime	Specifies the time interval after which the InQueue InterruptIntercept announcement (if one is provisioned) will be played to callers in the MLHG's queue. Note that the default value of 0 indicates that the announcement shall never be played.	integer	0, 30..180	N/A
InQueueInterruptIntercept	Specifies the intercept announcement to be played to callers in the MLHG's queue periodically after the number of seconds specified in the InQueueInterruptTime.	0..15	Character string – must represent a valid intercept.[1]	N/A
IsRoutingAllowed	This attribute can be provisioned to allow or deny subscriber rerouting. If the MLHG is provisioned to allow subscriber rerouting, the subscriber rerouting will only take place for MLHG members that are provisioned for subscriber rerouting	Boolean	True, False	False
SendPilotDisplay	Controls whether the OpenScape Voice sends the Pilot's display information to the Hunt Group members for hunted calls	Boolean	True, False	False
isMlhgEnhancedRerouteAllowed	Controls whether Enhanced Subscriber Rerouting is allowed in case a distributed call to a Hunt Group member is rejected due to Call Admission Control. The default value for this parameter shall be disabled.	Boolean	True, False	False

Table 742 Modify Mlhg Info Input Fields (Sheet 4 of 5)

Field Name	Description	Length	Range/Values	Default Value
OverflowDnNoRerouting	Define san Overflow destination in case a distributed call to the Hunt Group member is rejected due to Call Admission Control or WAN Failure AND Subscriber Rerouting (Enhanced or Basic) either fails or is not allowed.	String		
AllowCallsToOverflow	Allow calls to overflow (Unconditional). Applies for Profile Only and SIP Master Pilot Hunt Group DNs.	Boolean	True, False	
AllowUnansweredCallsToRequeue	Applies for Profile Only and SIP Master Pilot Hunt Group DNs.	Boolean	True, False	
[1]The list of valid existing intercepts is available using the *GetInterceptList* SOAP request (refer to Section 7.8, “Get Intercept List”).				

Table 742 Modify Mlhg Info Input Fields (Sheet 5 of 5)

15.2.3 Delete Mlhg

This method deletes the specified Hunt Group.

Table 743 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.04.07		subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_MLHG_TERMS_EXIST subRC_APP_MLHG_TERM_DELETION_FAILED subRC_DB_SUB_DN_LOOKUP_FAILED subRC_DB_COMM_INTERNAL_RECOVERY_ERROR

Table 743 Delete Mlhg Return Codes

Table 744 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
BgName	Business Group Name. Mandatory if the PilotServiceId is a BGL.	See definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
PilotServiceId	The DN of the subscriber/BGL that represents the Pilot DN of the MLHG. That is, the number that when dialed will cause hunting through the MLHG members to take place.	See definition of ServiceId in Section 5.4, “Primitive Data Definitions”, on page 66.		N/A
RemoveMlhg Terms	Indicates that all existing terminals in this MLHG shall be removed from the MLHG. If any terminals exist and this attribute is false, the MLHG deletion is rejected with an appropriate error message to the operator.	Boolean	True, False	False

Table 744 Delete Mlhg Input Fields

15.2.4 Get Mlhg Info

This method retrieves information about a Multi-Line Hunt Group.

Table 745 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.04.07		subRC_APP_MLHG_PILOT_DN_LEN_INVALID subRC_APP_MLHG_PILOT_DN_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_MLHG_PILOT_DN_DOES_NOT_EXIST subRC_DB_MLHG_PILOT_DN_LOOKUP_FAILED subRC_APP_MLHG_PILOT_DN_BG_ID_CONFLICT subRC_DB_BG_LOOKUP_FAILED

Table 745 Get Mlhg Info Return Codes

Table 746 lists the *input* fields for this method.

Field Name	Description	Length	Range/Value	Default Value
BgName	Business Group Name. Mandatory if the PilotServiceId is a BGL.	See definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
BOName	Branch Office Name to be associated with MLHG V7.00.01.09	char	0…30	Must coincide with BOName of the Pilot DN.
RepresentativeEndpointName	V7.00.01.09 The representative endpoint is the endpoint that is configured as the associated endpoint for all subscribers and endpoints of the branch office. With this association, the subscribers and endpoints of the branch office are logically tied together. For branch offices with a survivable proxy, following the proxy registration model, the representative endpoint is the configured survivable proxy endpoint. For branch offices with a survivable gateway, following the dual registration model, the representative is the configured survivable gateway endpoint.	char	1-16	Optional
AssociatedEndpointIP	V7.00.01.09 The associated Endpoint IP Address	char	4 octets	Optional

Table 746 Get Mlhg Info Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Value	Default Value
PilotServiceId	The DN of the subscriber/BGL that represents the Pilot DN of the MLHG. That is, the number that when dialed will cause hunting through the MLHG members to take place.	See definition of ServiceId in Section 5.4, "Primitive Data Definitions", on page 66.		N/A
GetMlhgInfo OptionList	Currently, the only option supported is IncludeMlhgTermList, which indicates that the list of MLHG terminals and their attributes shall be displayed along with the MLHG info.	Array of enum	IncludeMlhgTermList	Empty array
InterfaceWSDL	Refer to Section 5.1.1, "The Interface WSDL and Interface Release Control", on page 61.			
IsRoutingAllowed	This attribute can be provisioned to allow or deny subscriber rerouting. If the MLHG is provisioned to allow subscriber rerouting, the subscriber rerouting will only take place for MLHG members that are provisioned for subscriber rerouting	Boolean	True, False	False

Table 746 Get Mlhg Info Input Fields (Sheet 2 of 2)

Table 747 lists the *output* fields for this method.

Field Name	Description	Length	Range/Value	Default Value
BgName	Business Group Name. Mandatory if the DN to be used as the Pilot DN is a BGL. For more information, refer to Section 5.4, "Primitive Data Definitions", on page 66.			
OperatorId	Refer to Section 5.4, "Primitive Data Definitions", on page 66.			
PilotServiceId	The DN of the subscriber/BGL that represents the Pilot DN of the MLHG – that is, the number that when dialed will cause hunting through the MLHG members to take place.		See definition of ServiceId in Section 5.4, "Primitive Data Definitions", on page 66.	N/A

Table 747 Get Mlhg Info Output Fields (Sheet 1 of 4)

Field Name	Description	Length	Range/Value	Default Value
HuntType	The type of hunting algorithm used by the MLHG to determine which MLHG member should be presented with incoming calls to the Pilot DN. Notes: • *CircularWithHunt*: is more properly referred to as CircularWithMemory. • *UCD*: Uniform Call Distribution. This indicates that an ordered list of idle terminals is maintained and calls are presented to those that have been idle the longest. • *Manual*: Indicates that calls are not presented automatically; instead, agents must view calls via the CallManager application and manually select which calls to process. • *Parallel:* All available members of the Hunt Group are alerted simultaneously whenever a new call arrives at the Hunt Group	enum	Linear CircularWithHunt UCD Manual Parallel	Linear
OverFlowDN	Specifies the destination of unanswered incoming calls when the queue is full. This DN can be an E.164DN or a private extension if this MLHG is part of a BG.		See definition of ServiceId in Section 5.4, “Primitive Data Definitions”, on page 66.	“”
ConfQueue Size	Indicates the number of calls that may be queued waiting for an available phone (terminal).	integer	0..MaxQueueSize	N/A
MaxQueue Size	Indicates the maximum number of calls that may be queued waiting for an available phone (terminal). This parameter can be configured by the telco and sets a hard limit of the queue size.	integer	0..511	N/A
CallingParty Source	Indicates which calling number should be used on outgoing calls from a device that is a terminal in the MLHG.	enum	PilotDN UseLineAndPilotDN	N/A
NoAnsAdv Time	Indicates the time for which a terminal will ring before the hunting algorithm moves on to continue searching for the next available terminal.	integer	5..3600 (seconds)	N/A

Table 747 Get Mlhg Info Output Fields (Sheet 2 of 4)

Field Name	Description	Length	Range/Value	Default Value
Intercept Annoncement Id	Indicates the configured intercept announcement that will be played when an incoming call is queued because no terminals are available.	15	Character string; must represent a valid intercept.[1]	N/A
NightService DN	The directory number to which calls are diverted when the Night Service feature is active.		See definition of ServiceId in Section 5.4, "Primitive Data Definitions", on page 66.	N/A
AutoMakeBusy	Indicates whether DNs in MLHG are marked busy on auto advance timer expiry.	Boolean	True, False	N/A
MaxTimeIn Queue	Monitors the time in queue for calls to the MLHG. When the actual time exceeds the provisioned value and other conditions are met, the queued call is removed and routed to the Night Service DN.	integer	0..600	N/A
NightService ManualStatus	Indicates whether Manual Night Service is currently active.	Boolean	True, False	N/A
NightService AutomaticStatus	Display only. Indicates whether Automatic Night Service is currently active.	Boolean	True, False	N/A
PostCallTimer	Number of seconds to pause after each call before presenting next call to same agent.	integer	0..120	N/A
TMStatus	Indicates whether traffic measurements are currently enabled for this MLHG.	Boolean	True, False	N/A
InQueueInterruptTime	Specifies the time interval after which the InQueue InterruptIntercept announcement (if one is provisioned) will be played to callers in the MLHG's queue. Note that the default value of 0 indicates that the announcement shall never be played.	integer	0, 30..180	N/A
InQueueInterruptIntercept	Specifies the intercept to be played to callers in the MLHG's queue periodically after the number of seconds specified in the InQueueInterruptTime.	0..15	Character string – must represent a valid intercept.[1]	N/A

Table 747 Get Mlhg Info Output Fields (Sheet 3 of 4)

Field Name	Description	Length	Range/Value	Default Value
IsReroutingAllowed	This attribute can be provisioned to allow or deny subscriber rerouting. If the MLHG is provisioned to allow subscriber rerouting, the subscriber rerouting will only take place for MLHG members that are provisioned for subscriber rerouting	Boolean	True, False	False
SendPilotDisplay	Controls whether the OpenScape Voice sends the Pilot's display information to the Hunt Group members for hunted calls	Boolean	True, False	False
isMlhgEnhancedRerouteAllowed	Controls whether Enhanced Subscriber Rerouting is allowed in case a distributed call to a Hunt Group member is rejected due to Call Admission Control. The default value for this parameter shall be disabled.	Boolean	True, False	False
OverflowDnNoRerouting	Define san Overflow destination in case a distributed call to the Hunt Group member is rejected due to Call Admission Control or WAN Failure AND Subscriber Rerouting (Enhanced or Basic) either fails or is not allowed.	String		
[1]The list of valid existing intercepts is available using the *GetInterceptList* SOAP request (refer to Section 7.8, "Get Intercept List").				

Table 747 Get Mlhg Info Output Fields (Sheet 4 of 4)

15.2.5 Add Sub To Mlhg

This method adds a subscriber to a Hunt Group.

Table 748 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.04.07		subRC_SUCCESS subRC_APP_BG_NAME_LEN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_MLHG_PILOT_DN_LEN_INVALID subRC_APP_MLHG_PILOT_DN_INVALID subRC_APP_SUB_LEN_INVALID subRC_APP_SUB_DN_INVALID subRC_APP_SUB_DN_DOES_NOT_EXIST subRC_DB_SUB_DN_LOOKUP_FAILED subRC_APP_SUB_DN_DOES_NOT_EXIST subRC_DB_SUB_DN_LOOKUP_FAILED subRC_APP_MLHG_TERM_H323_INVALID subRC_APP_MLHG_TERM_PARENT_TEEN_CONFLICT subRC_APP_MLHG_PILOT_DN_SUB_BG_CONFLICT subRC_APP_MLHG_MEMBER_FEATURE_CONFLICT subRC_DB_MLHG_TERM_LOOKUP_FAILED subRC_APP_MLHG_TERM_IN_USE subRC_DB_MLHG_PILOT_DN_TERM_LIST_LOOKUP_FAILED subRC_APP_MLHG_TERMS_NUMBER_OUT_OF_RANGE subRC_APP_MLHG_PILOT_DN_DOES_NOT_EXIST subRC_DB_MLHG_PILOT_DN_LOOKUP_FAILED subRC_DB_MLHG_PILOT_DN_MISSING subRC_DB_MLHG_TERM_LOOKUP_FAILED subRC_APP_MLHG_TERM_CSTA_NOT_SUBSCRIBED subRC_DB_SUB_OFFICE_CODE_LOOKUP_FAILED subRC_DB_SUB_HOME_DN_LOOKUP_FAILED subRC_APP_SUB_DISCONNECTED subRC_APP_MLHG_PILOT_DN_CSTA_NOT_SUBSCRIBED subRC_APP_MLHG_QUEUE_PRIORITY_INVALID subRC_DB_COMM_RTM_LOOKUP_FAILED subRC_DB_XDM_EP_LOOKUP_FAILED subRC_APP_MLHG_EP_TYPE_UNSUPPORTED subRC_DB_MLHG_TERM_INSERTION_FAILED subRC_DB_COMM_INTERNAL_RECOVERY_ERROR subRC_DB_MLHG_PRIMARY_PILOT_DN_LOOKUP_FAILED subRC_APP_JOB_ADD_FAILED subRC_WAR_MLHG_TERM_INSERTION_DATA_IGNORED

Table 748 Add Sub To Mlhg Return Codes

Table 749 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
BgName	Business Group Name. Mandatory if the PilotServiceId is a BGL.	See definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			

Table 749 Add Sub To Mlhg Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
PilotServiceId	The DN of the subscriber/BGL that represents the Pilot DN of the MLHG. That is, the number that when dialed will cause hunting through the MLHG members to take place.	See definition of ServiceId in Section 5.4, "Primitive Data Definitions", on page 66.		N/A
ServiceId	The DN of the existing subscriber/ BGL to be added as a new MLHG terminal.	See definition of ServiceId in Section 5.4, "Primitive Data Definitions", on page 66.		N/A
Position	Indicates the position in the MLHG in which the new terminal should be inserted. If not provided, the terminal is added in the last position. That is, one greater than the currently existing last terminal.	integer		Position of current last terminal + 1
MakeBusySubControlled	Indicates whether the MLHG Member can control the Make Busy functionality for the terminal.	Boolean	True, False	False
StopHuntSub Controlled	Indicates whether the MLHG Member can control the Stop Hunt functionality for the terminal.	Boolean	True, False	False
MlhgBusy Status	Indicates the initial Make Busy state for the terminal.	enum	NotBusy Busy	NotBusy
MlhgStopHuntStatus	Indicates the initial Stop Hunt state for the terminal.	enum	NotStopHunt StopHunt	NotStopHunt
MlhgAttendant	Indicates whether member is an attendant (not used in OpenScape Voice, only in CallManager).	Boolean	True, False	False
QueuePriority	Specifies the order in which this MLHG member shall search queues, assuming the terminal is a member of multiple MLHGs. This value is stored on a per MLHG/per MLHG member basis so a terminal has a separate value for each MLHG of which it is a member. Queues with lower number priorities are checked before higher numbered priority queues. Queues with the same priority may be checked in any order. Optional. If not provided, the default value 255 (lowest priority) is used.	integer	1..255	255

Table 749 Add Sub To Mlhg Input Fields (Sheet 2 of 2)

15.2.6 Create Mlhg Term

This method creates a Hunt Group terminal. Not currently supported.

Table 750 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
N/A		subRC_UNSUPPORTED_FUNCTION

Table 750 Create Mlhg Term Return Codes

15.2.7 Modify Mlhg TermInfo

This method adds, removes, and modifies Hunt Group terminal information.

Table 751 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.04.07		subRC_SUCCESS subRC_APP_SUB_LEN_INVALID subRC_APP_SUB_DN_INVALID subRC_APP_MLHG_PILOT_DN_LEN_INVALID subRC_APP_MLHG_PILOT_DN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_SUB_DN_DOES_NOT_EXIST subRC_DB_SUB_DN_LOOKUP_FAILED subRC_APP_MLHG_PILOT_DN_DOES_NOT_EXIST subRC_DB_MLHG_PILOT_DN_LOOKUP_FAILED subRC_DB_MLHG_LOOKUP_FAILED subRC_APP_MLHG_TERM_NOT_IN_MLHG subRC_DB_MLHG_TERM_LOOKUP_FAILED subRC_APP_MLHG_PILOT_DN_LEN_INVALID subRC_APP_MLHG_PILOT_DN_INVALID subRC_APP_MLHG_PRIMARY_PILOT_DN_NOT_ALLOWED subRC_APP_MLHG_PRIMARY_PILOT_DN_DOES_NOT_EXIST subRC_DB_MLHG_PRIMARY_PILOT_DN_LOOKUP_FAILED subRC_APP_MLHG_DOES_NOT_EXIST subRC_DB_MLHG_TERM_LOOKUP_FAILED subRC_APP_MLHG_TERM_NOT_IN_MLHG subRC_DB_SUB_DN_LOOKUP_FAILED subRC_APP_MLHG_MEMBER_MODIFY_FAILED subRC_APP_JOB_ADD_FAILED subRC_DB_MLHG_TERM_LOOKUP_FAILED subRC_APP_MLHG_PILOT_DN_CSTA_NOT_SUBSCRIBED subRC_APP_MLHG_QUEUE_PRIORITY_INVALID subRC_DB_MLHG_TERM_INSERTION_FAILED

Table 751 Modify Mlhg Term Info Return Codes

Table 752 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
BgName	Business Group Name. Mandatory if the PilotServiceId is a BGL.	See definition of BGName in Section 5.4, "Primitive Data Definitions", on page 66.		
OperatorId	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.			
BOName	Branch Office Name to be associated with MLHG V7.00.01.09	char	0…30	Must coincide with BOName of the Pilot DN.
PilotServiceId	The DN of the subscriber/BGL that represents the Pilot DN of the MLHG. That is, the number that when dialed will cause hunting through the MLHG members to take place.	See definition of ServiceId in Section 5.4, "Primitive Data Definitions", on page 66.		N/A
Member ServiceId	The DN of the MLHG terminal.	See definition of ServiceId in Section 5.4, "Primitive Data Definitions", on page 66.		N/A
PrimaryPilot ServiceId	Assuming the terminal is a member of multiple MLHGs, this parameter indicates that the terminal's primary MLHG shall be changed to the MLHG of the Pilot DN provided.	See definition of ServiceId in Section 5.4, "Primitive Data Definitions", on page 66.		N/A
Position	Indicates the position in the MLHG to which the terminal should be moved.	integer		N/A
MakeBusySubControlled	Indicates whether the MLHG Member can control the Make Busy functionality for the terminal.	Boolean	True, False	N/A
StopHuntSub Controlled	Indicates whether the MLHG Member can control the Stop Hunt functionality for the terminal.	Boolean	True, False	N/A
MlhgBusy Status	Indicates the Make Busy state for the terminal.	enum	NotBusy Busy	N/A
MlhgStopHuntStatus	Indicates the Stop Hunt state for the terminal.	enum	NotStopHunt StopHunt	N/A
MlhgAttendant	Indicates whether member is an attendant (not used in OpenScape Voice, only in CallManager).	Boolean	True, False	N/A

Table 752 Modify Mlhg Term Info Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
QueuePriority	Specifies the order in which this MLHG member shall search queues, assuming the terminal is a member of multiple MLHGs. This value is stored on a per MLHG/per MLHG member basis, so a terminal has a separate value for each MLHG of which it is a member. Queues with lower number priorities are checked before higher number priority queues. Queues with the same priority may be checked in any order.	integer	1..255	N/A

Table 752 Modify Mlhg Term Info Input Fields (Sheet 2 of 2)

15.2.8 Delete Mlhg Term

This method removes the specified Hunt Group terminal from the Hunt Group whose Pilot DN was specified in the request. The DN remains an ordinary subscriber/BGL after this request is performed, and it may still be a member of other Hunt Groups.

Table 753 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
04.04.07		subRC_INTERNAL_ERROR subRC_MLHG_TERM_DEL_ERR subRC_INVLD_BG_NAME subRC_INVLD_FEATURE_DATA subRC_INVLD_SUB_DN subRC_SUB_DOES_NOT_EXIST subRC_SUCCESS subRC_INTERNAL_RECOVERY_ERR

Table 753 Delete Mlhg Term Return Codes

Table 754 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
BgName	Business Group Name. Mandatory if the PilotServiceId is a BGL.	See definition of BGName in Section 5.4, “Primitive Data Definitions”, on page 66.		
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			

Table 754 Delete Mlhg Term Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
PilotServiceId	The DN of the subscriber/BGL that represents the Pilot DN of the MLHG. That is, the number that when dialed will cause hunting through the MLHG members to take place.	See definition of ServiceId in Section 5.4, “Primitive Data Definitions”, on page 66.		N/A
Member ServiceId	The DN of the MLHG terminal.	See definition of ServiceId in Section 5.4, “Primitive Data Definitions”, on page 66.		N/A

Table 754 Delete Mlhg Term Input Fields (Sheet 2 of 2)

15.2.9 Get Mlhg TM Data

This method retrieves Hunt Group terminal data.

Table 755 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
10.00.01.19		subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_MLHG_TMDATA_DIRECTORY_INVALID subRC_APP_MLHG_TMDATA_HL_DIRECTORY_OPEN_FAILED

Table 755 Get Mlhg TM Data Return Codes

Table 756 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
StartDirName	A directory with which to start. The directory name is compared against each directory present in the root traffic measurement directory (/global/user/oms) and only directories created after the timestamp of the provided input directory are returned. The expected format for the input directory: “SRX3000-YYYYMMDDTHHMMSS”	1..25	Character string	“”

Table 756 Get Mlhg TM Data Input Fields

16 Endpoint Profiles

Endpoint Profiles are used for SIP Network Servers such as SIP Gateways (access to the PSTN, HiPath 4000, HiPath 3000), SIP Proxy Servers (Comdasys), SIP Voice Mail Server (Xpressions), and SIP Conference Bridge. These profiles allow services and feature settings to be defined for these endpoints that need specific services.

16.1 Create Ep Profile

This API is used to Create Endpoint Profiles for SIP Network Servers.

Table 757 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
05.03.01		subRC_SUCCESS subRC_APP_EP_NAME_LEN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_EP_PROFILE_INVALID subRC_APP_EP_PROFILE_EXIST subRC_DB_EP_PROFILE_LOOKUP_FAILED subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_APP_EP_PRI_SUP_INVALID subRC_APP_UTIL_LANG_LEN_INVALID subRC_APP_UTIL_LANG_NOT_ALLOWED subRC_APP_UTIL_LANG_INVALID subRC_APP_EP_VOIP_MISSING subRC_APP_EP_T38_ECH_CAN_CONFLICT subRC_APP_EP_56QOS_INVALID subRC_APP_EP_QOS_QOSPROFILE_CONFLICT subRC_APP_QOS_PROFILE_LEN_INVALID subRC_APP_EP_QOSPROFILE_INVALID subRC_APP_EP_QOSPRIM_MISSING subRC_APP_EP_QOSPRIM_PACK_PER_INVALID subRC_APP_EP_QOSCLASS_QOSVALUE_INVALID subRC_APP_EP_QOSSEC_PACK_PER_INVALID subRC_APP_EP_QOSSEC_TOS_INVALID subRC_APP_EP_QOSTHRD_QOSSEC_CONFLICT subRC_APP_EP_QOSTHRD_QOS4TH_CONFLICT subRC_APP_EP_QOS4TH_QOS5TH_CONFLICT subRC_APP_EP_QOS5TH_QOS6TH_CONFLICT subRC_APP_EP_QOSTHRD_PACK_PER_INVALID subRC_APP_EP_QOSTHRD_TOS_INVALID subRC_APP_EP_QOS4TH_PACK_PER_INVALID subRC_APP_EP_QOS4TH_TOS_INVALID subRC_APP_EP_QOS5TH_PACK_PER_INVALID subRC_APP_EP_QOS5TH_TOS_INVALID subRC_APP_EP_QOS6TH_PACK_PER_INVALID subRC_APP_EP_QOS6TH_TOS_INVALID subRC_DB_EP_INSERTION_FAILED

Table 757 Create Ep Profile Return Codes

Table 758 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
EpType	Type of Endpoint Profile.	enum	Network_server	Network_server
EpName	Name of Endpoint Profile to create.	1..30	Character string	required

Table 758 Create Ep Profile Input Fields (Sheet 1 of 4)

Field Name	Description	Length	Range	Default Value
EpMgtInfo				
(EpMgtInfo) BGName	Name of the Business Group associated with the endpoint. Note that when a BG is provided, the BG's Private Numbering Plan, if any, is used for translation of incoming calls through this Endpoint Profile.	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.		""
(EpMgtInfo) ClassOfService	The name of the class of service.	Refer to the definition of COS in Section 9.1, "Create Subscriber", on page 303.		""
(EpMgtInfo) RateArea	If the Endpoint Profile has no Rate Area assigned, it will only be possible to access default routes.The available list of Rate Areas is retrievable using the GetRateAreaList interface method.	Refer to the definition of RateArea in Section 9.1, "Create Subscriber", on page 303.		""
(EpMgtInfo) LataDigits	Identifies the LATA to which the Endpoint Profile belongs. (The switch may serve subscribers across multiple latas.) This information is used during SCP 800 number translations to an SCP.	0..3	Numeric string	""
(EpMgtInfo) PrivacySupport	Privacy settings for a subscriber.	enum	SipBasic, SipFull	SipBasic
(EpMgtInfo) Language	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.			"System Default"
(EpMgtInfo) CallingLoc	Indicates the Endpoint Profile's location. Used for routing emergency calls to the answering point that serves the Calling Location.	1..15	Any existing Calling Location.[1]	""
(EpMgtInfo) FailedCallsInterceptTreatment	Used to indicate the treatment for unsuccessful calls. If the Intercept Treatment is set to True (Yes), unsuccessful calls will be sent to intercept treatments, either play tones or announcement, depends on the switch configuration. If the Intercept Treatment is set to False (No), unsuccessful calls will be released with the release cause. The default value of the Intercept Treatment is True (Yes). It applies to all markets and solutions, i.e. it applicable to ANSI, ETSI, Enterprise.	Boolean	True, False	True

Table 758 Create Ep Profile Input Fields (Sheet 2 of 4)

Field Name	Description	Length	Range	Default Value
(EpMgtInfo) NumberPlanName	The name of the Private Numbering Plan to associate with this Endpoint Profile. If none is provided, the BG's numbering plan applies. For EPs not associated with a BG, only the system default numbering plan applies.	1..31	Character string.	“”
(EpMgtInfo) TimeZone	Specifies the endpoint profile's time zone and determines how the time-dependent services work. For a list of valid time zones, refer to Appendix A, “Time Zones”.		Any valid time zone.	“Local“
(EpMgtInfo) ImpactLevel	The security zone of an endpoint. V7.00.15.0 forward.	Enum	• ImpactLevelZero • ImpactLevelOne • ImpactLevelTwo • ImpactLevelThree • ImpactLevelFour • ImpactLevelFive • ImpactLevelSix • ImpactLevelSeven • ImpactLevelEight • ImpactLevelNine • ImpactLevelUnclassified	
Voip				
(Voip) Bearer_ Speech	See Section 5.4, “Primitive Data Definitions”, on page 66.			True
(Voip) Bearer_3_1K	See Section 5.4, “Primitive Data Definitions”, on page 66.			False
(Voip) Bearer_56K	See Section 5.4, “Primitive Data Definitions”, on page 66.			False
(Voip) Bearer_64K	See Section 5.4, “Primitive Data Definitions”, on page 66.			False
(Voip) FaxMode	See Section 5.4, “Primitive Data Definitions”, on page 66.			G711
(Voip) Echo Cancellation	See Section 5.4, “Primitive Data Definitions”, on page 66.			True
(Voip) Silence Suppression	See Section 5.4, “Primitive Data Definitions”, on page 66.			False
(Voip) Fax Statistics	See Section 5.4, “Primitive Data Definitions”, on page 66.			False

Table 758 Create Ep Profile Input Fields (Sheet 3 of 4)

Field Name	Description	Length	Range	Default Value
(Voip) Outband DTMF Signaling	See Section 5.4, “Primitive Data Definitions”, on page 66.			False
(Voip) Suppress OutbandDTMFWith G711	See Section 5.4, “Primitive Data Definitions”, on page 66.			False
ListOf EpFeatures	The list of features that can be assigned to the End Point profile.		1. **CfCISNAME**; Called Party Name Delivery 2. **CfCISNUM**; Called Party Number Delivery 3. **CfEpCNAM**; Incoming Calling Name Delivery for Endpoint 4. **CfTRS**; Toll restrictions 5. **CfEpVM**; Voice Mail for Endpoint 6. **CfEpCT**; Call Transfer for Endpoint 7. **CfCFID;** (Call Forward Invalid Destination 8. **CfNWCPS**; Network Wide Call Park to Serv	N/A
[1]The available list of Calling Locations can be retrieved using the *GetCallingLocationList* interface method (see Section 7.9, “Get Calling Location List”).				

Table 758 Create Ep Profile Input Fields (Sheet 4 of 4)

16.1.1 Update Ep Mgt Info

This method changes management attributes of the endpoint profile.

Table 759 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
05.03.01		subRC_SUCCESS subRC_APP_EP_NAME_LEN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_EP_PROFILE_DOES_NOT_EXIST subRC_DB_EP_PROFILE_LOOKUP_FAILED subRC_APP_BG_NAME_LEN_INVALID subRC_APP_BG_DOES_NOT_EXIST subRC_DB_BG_LOOKUP_FAILED subRC_DB_EP_MGTINFO_COS_LOOKUP_FAILED subRC_APP_EP_MGTINFO_COS_DOES_NOT_EXIST subRC_DB_EP_MGTINFO_RATE_AREA_LOOKUP_FAILED subRC_APP_EP_MGTINFO_RATE_AREA_DOES_NOT_EXIST subRC_DB_EP_MGTINFO_CALLING_LOC__LOOKUP_FAILED subRC_APP_EP_MGTINFO_CALLING_LOC_DOES_NOT_EXIST subRC_APP_EP_MGTINFO_PRIVACY_SUPPORT_INVALID subRC_APP_UTIL_LANG_LEN_INVALID subRC_APP_UTIL_LANG_NOT_ALLOWED subRC_APP_UTIL_LANG_INVALID subRC_DB_EP_MODIFY_FAILED

Table 759 Update Ep Mgt Info Return Codes

Table 760 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
EpName	The name of an existing Endpoint Profile.	1..30	Character string.	required
NewEpName	The New Name of the Endpoint Profile	1…30	Character String	Optional
OperatorId	Refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
EpMgtInfo				
(EpMgtInfo) BGName	Refer to Section 16.1, “Create Ep Profile”, on page 644.			N/A
(EpMgtInfo) ClassOfService	Refer to Section 16.1, “Create Ep Profile”, on page 644.			N/A
(EpMgtInfo) RateArea	Refer to Section 16.1, “Create Ep Profile”, on page 644.			N/A
(EpMgtInfo) LataDigits	Refer to Section 16.1, “Create Ep Profile”, on page 644.			N/A
(EpMgtInfo) PrivacySupport	Refer to Section 16.1, “Create Ep Profile”, on page 644.			N/A
(EpMgtInfo) Language	Refer to Section 16.1, “Create Ep Profile”, on page 644.			N/A
(EpMgtInfo) BGName	Refer to Section 16.1, “Create Ep Profile”, on page 644.			N/A

Table 760 Update Ep Mgt Info Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
(EpMgtInfo) ClassOfService	Refer to Section 16.1, "Create Ep Profile", on page 644.			N/A
(EpMgtInfo) CallingLoc	Indicates the Endpoint Profile's location. Used for routing emergency calls to the answering point that serves the Calling Location.	1..15	Any existing Calling Location.[1]	N/A
(EpMgtInfo) FailedCallsInterceptTreatment	Used to indicate the treatment for unsuccessful calls. If the Intercept Treatment is set to True (Yes), unsuccessful calls will be sent to intercept treatments, either play tones or announcement, depends on the switch configuration. If the Intercept Treatment is set to False (No), unsuccessful calls will be released with the release cause. The default value of the Intercept Treatment is True (Yes). It applies to all markets and solutions, i.e. it applicable to ANSI, ETSI, Carrier, Enterprise, CarrierNoSS7.	Boolean	True, False	N/A
(EpMgtInfo) NumberPlanName	The name of the Private Numbering Plan to associate with this Endpoint Profile. In order to remove an existing Number Plan Name assignment, the parameter can be sent as an empty string. Thereafter, the BG's numbering plan applies. If the parameter is not sent, no change is made to the existing assignment,	Refer to Section 16.1, "Create Ep Profile", on page 644.		N/A
(EpMgtInfo) TimeZone	Specifies the endpoint profile's time zone and determines how the time-dependent services work. For a list of valid time zones, refer to Appendix A, "Time Zones".		Any valid time zone.	N/A
(EpMgtInfo) ImpactLevel	The security zone of an endpoint. V7.00.15.0 forward.	Enum	ImpactLevelZero ImpactLevelOne ImpactLevelTwo ImpactLevelThree ImpactLevelFour ImpactLevelFive ImpactLevelSix ImpactLevelSeven ImpactLevelEight ImpactLevelNine ImpactLevelUnclassified	

Table 760 Update Ep Mgt Info Input Fields (Sheet 2 of 2)

16.1.2 Update Ep Capabilities

This method changes the capabilities of the endpoint.

Table 761 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
05.03.01		subRC_SUCCESS subRC_APP_EP_NAME_LEN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_EP_PROFILE_DOES_NOT_EXIST subRC_DB_SUB_DN_LOOKUP_FAILED subRC_APP_EP_VOIP_DELETION_FAILED subRC_APP_EP_T38_ECH_CAN_CONFLICT subRC_DB_EP_MODIFY_FAILED

Table 761 Update Ep Capabilities Return Codes

Table 762 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
EpName	The name of an existing Endpoint Profile.	1..30	Character string.	required
OperatorId	Refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
Voip				
(Voip) Bearer_Speech	Refer to Section 16.1, “Create Ep Profile”, on page 644.			N/A
(Voip) Bearer_3_1K	Refer to Section 16.1, “Create Ep Profile”, on page 644.			N/A
(Voip) Bearer_56K	Refer to Section 16.1, “Create Ep Profile”, on page 644.			N/A
(Voip) Bearer_64K	Refer to Section 16.1, “Create Ep Profile”, on page 644.			N/A
(Voip) FaxMode	Refer to Section 16.1, “Create Ep Profile”, on page 644.			N/A
(Voip) EchoCancellation	Refer to Section 16.1, “Create Ep Profile”, on page 644.			N/A
(Voip) Silence Suppression	Refer to Section 16.1, “Create Ep Profile”, on page 644.			N/A
(Voip) FaxStatistics	Refer to Section 16.1, “Create Ep Profile”, on page 644.			N/A
(Voip) Outband DTMFSignaling	Refer to Section 16.1, “Create Ep Profile”, on page 644.			N/A

Table 762 Update Ep Capabilities Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
(Voip) Suppress OutbandDTMF WithG711	Refer to Section 16.1, “Create Ep Profile”, on page 644.			N/A

Table 762 Update Ep Capabilities Input Fields (Sheet 2 of 2)

16.1.3 Update Ep Features

This method adds, removes and replaces service features.

Table 763 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
05.03.01		subRC_SUCCESS subRC_APP_EP_NAME_LEN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_EP_PROFILE_DOES_NOT_EXIST subRC_DB_EP_PROFILE_LOOKUP_FAILED subRC_DB_EP_MODIFY_FAILED

Table 763 Update Ep Features Return Codes

Table 764 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
EpName	The name of an existing Endpoint Profile.	1..30	Character string.	required
OperatorId	Refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
ListofFeatures	Refer to Section 16.1, “Create Ep Profile”, on page 644.		• **CfTRS**; Toll restrictions • **CfEpVM**; Voice Mail • **CfEpCT**; Call Transfer • **CfCFID**; Call Forward Invalid Destination • **CfCNIM;** CSTA External device name provider **(V7.00.01.10)**	N/A

Table 764 Update Ep Features Input Fields

16.1.4 Get Ep Info

This method retrieves the endpoint profile.

Table 765 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
05.03.01	N/A	subRC_SUCCESS subRC_APP_EP_PROFILE_LEN_INVALID subRC_APP_EP_PROFILE_DOES_NOT_EXIST subRC_DB_EP_PROFILE_LOOKUP_FAILED subRC_DB_BG_LOOKUP_FAILED subRC_DB_UTIL_LANG_LOOKUP_FAILED subRC_DB_FEAT_DATA_LOOKUP_FAILED

Table 765 Get Ep Info Return Codes

Table 766 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default Value
EpName	The name of an existing Endpoint Profile.	1..30	Character string.	required

Table 766 GetEpInfo Input Fields

Table 767 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
EpType	Refer to Section 16.1, "Create Ep Profile", on page 644.			N/A
EpMgtInfo				
(EpMgtInfo) BGName	Name of the Business Group associated with the endpoint.	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.		N/A
(EpMgtInfo) ClassOfService	The name of the class of service.	Refer to the definition of COS in Section 9.1, "Create Subscriber", on page 303.		N/A
(EpMgtInfo) RateArea	If the subscriber has no Rate Area assigned, the subscriber will only be able to access default routes.The available list of Rate Areas is retrievable using the GetRateAreaList interface method.	Refer to the definition of RateArea in Section 9.1, "Create Subscriber", on page 303.		N/A
(EpMgtInfo) PrivacySupport	Privacy settings for a subscriber.	enum	SipBasic, SipFull	N/A
(EpMgtInfo) Language	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.			N/A
(EpMgtInfo) ImpactLevel	The security zone of an endpoint. V7.00.15.0 forward.	Enum	• ImpactLevelZero • ImpactLevelOne • ImpactLevelTwo • ImpactLevelThree • ImpactLevelFour • ImpactLevelFive • ImpactLevelSix • ImpactLevelSeven • ImpactLevelEight • ImpactLevelNine • ImpactLevelUnclassified	
ListOf EpFeatures	The list of features that can be assigned to the End Point profile.		• CfTRS; Toll and Call Restrictions • CfVM; Call Forwarding Voice Mail • EpCT; Call Transfer for Endpoint	N/A
Subscriber AuditInfo. InitialProvTOY	The date and time stamp when the end point profile was created.	Please refer to the definition of TOY in Section 5.4, "Primitive Data Definitions", on page 66.		N/A

Table 767 Get Ep Info Output Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
Subscriber AuditInfo.Last OpProvTOY	The date and time stamp when the end point profile was last modified.	Please refer to the definition of TOY in Section 5.4, “Primitive Data Definitions”, on page 66.		N/A
Subscriber AuditInfo. LastOpId	The ID of the last operator to change the profile.	Please refer to the definition of OperatorId in Section 5.4, “Primitive Data Definitions”, on page 66.		N/A

Table 767 Get Ep Info Output Fields (Sheet 2 of 2)

16.1.5 Get Ep Info By Release

This method retrieves EP info.

Table 768 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
05.03.01		The return codes for this request are the same as those listed for Get Ep Info in Section 16.1.4, “Get Ep Info”.

Table 768 Get Ep Info By Release Return Codes

Table 769 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default Value
Interface Release	The Interface Release version to use. This provides information for older releases in their older format.	enum	Latest, Release6, Release8, Release9, Release10	Latest
EpName	The name of an existing Endpoint Profile.	1..30	Character string.	required

Table 769 Get Ep Info By Release Input Fields

The *output* fields for this method are the same as those described Get Ep Info in Table 767.

16.1.6 Get EP List

This method retrieves a list of the Endpoint Profiles existing on the switch. For each entry, the EP Name, EP Type and the BG Name are returned.

Table 770 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
05.03.01	N/A	subRC_SUCCESS subRC_DB_EP_PROFILE_NUM_LOOKUP_FAILED subRC_DB_EP_PROFILE_LOOKUP_FAILED subRC_DB_EP_PROFILE_TYPE_INVALID

Table 770 Get EP List Return Codes

Table 771 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required
InterfaceWSDL	The WSDL version of the request. This allows the SOAP Server to avoid returning fields that were introduced after the WSDL version in question.	Refer to Section 5.1.1, “The Interface WSDL and Interface Release Control”, on page 61.		N/A
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.			N/A
BGName	The name of the Business Group on which to filter.	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.		N/A
NumberPlanName	The name of the Private Numbering Plan on which to filter. Only BGs that are associated with this Number Plan will be returned.	1..31	Character string.	N/A

Table 771 Get BG List Input Field Descriptions

Table 772 lists the output fields for this method.

Field Name	Description	Length	Range	Default Value
unsp__GetEp ListResult (list)	A list of the endpoints including the following attributes for each list entry: EpName – The name of the end point EpType – The type associated with the end point. (Please see Section , “Endpoint Profiles”, on page 644.)			N/A

Table 772 Get EP List Output Fields

16.1.7 Delete Ep Profile

This method removes the endpoint profile.

Table 773 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
05.03.01	N/A	subRC_SUCCESS subRC_APP_EP_PROFILE_LEN_INVALID subRC_APP_EP_PROFILE_T,YPE_INVALID subRC_APP_EP_PROFILE_IN_USE subRC_DB_EP_PROFILE_REFERENCE_LOOKUP_FAILED

Table 773 Delete Ep Profile Return Codes

Table 774 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
EpName	The name of an existing Endpoint Profile.	1..30	Character string.	required
OperatorId	Refer to Section 5.4, “Primitive Data Definitions”, on page 66.			required

Table 774 Delete Ep Profile Input Fields

17 Background Job Processing Interfaces

17.1 Jobs

The OpenScape Voice system supports background jobs in specific situations. In future releases, it may be possible to schedule actions, such as scheduling the disconnection of a subscriber (for example, to occur at the end of the month), to occur later on the OpenScape Voice system. At this time, however, jobs are used for SOAP requests that require more processing time than a client will typically wait for.

There are several types of requests that require more time than available; the most common involves updates to Feature Profiles, which may require the updating of several thousand subscribers. Upon receipt of this type of request, the OpenScape Voice system returns the JobId tag in the ResultCodeStruct:. When this field is returned, it means that the request has been successfully received, validated, and applied to the Feature Profile and updates have been scheduled for the affected subscribers.

The status of the job may be obtained using the GetJobInfo method and a list of existing jobs may be obtained using the GetJobList method. The CancelJob method may be used to terminate a running or waiting job.

17.1.1 Get Job Info

This method retrieves information for a particular job.

Table 775 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
03.01.01		subRC_SUCCESS subRC_APP_JOB_ID_MISSING subRC_DB_JOB_ID_LOOKUP_FAILED subRC_APP_JOB_ID_DOES_NOT_EXIST

Table 775 Get Job Info Return Codes

Table 776 lists the *input* fields for this method.

Field Name	Description	Length	Range/Values	Default Value
JobId	The ID of a job in the format <operatorId>_<jobSeqenceNumber>.	49	Character string	N/A
StartSequence	The sequence number of the first JobResult entry that shall be returned.	int	1..1000	1
MaxSequence	The maximum number of JobResult entries that shall be returned.	int	1..1000	1000

Table 776 Get Job Info Input Fields

Table 777 lists the *output* fields for this method.

Field Name	Description	Length	Range/Values	Default Value
JobId	The ID of a job in the format <operatorId>_<jobSequenceNumber>.	49	Character string	N/A
JobComments	A text string optionally provided by the operator when a job is cancelled.	2000	Character string	N/A
OperatorID	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
JobStatus	The current status of the job.	enum	Waiting InProgress Complete Canceled Partial Error	N/A
JobAction	The action to be performed by this job.	enum	AddFeature RemoveFeature UpdateFeature UpdateBGDefaultQOSProfile UpdateQOSProfile Audit FeatureProfile CollectLogFiles GenericAuditHomeDn	N/A
ObjectType	The type of object for which this job has been submitted. FP is used to indicate a job to propagate features from a Feature Profile to its associated subscribers/BGLs.	enum	BG DN FP QP GENERIC	N/A
ObjectName	The ID of the FeatureProfile.	49	Numeric string	N/A
CreateDate	The timestamp when the job was created.	xsd__dateTime		N/A

Table 777 Get Job Info Output Fields (Sheet 1 of 2)

Field Name	Description	Length	Range/Values	Default Value
ExecutionDate	The timestamp when the job started execution.	xsd__dateTime		N/A
CancelDate	The timestamp when the job was cancelled, if at all.	xsd__dateTime		N/A
Max Sequences	The total number of JobResult entries existing for this job.	integer		N/A
Objects Processed	The number of objects processed. In the case of feature propagation from a Feature Profile to the associated subscribers/BGLs, the number of subscribers/BGLs processed.	integer		N/A
JobResultList	A list of job result entries, if any, where each entry consists of: • ObjectType • ObjectName • SequenceNumber • JobEntryResultCode	Array of structures		N/A

Table 777 Get Job Info Output Fields (Sheet 2 of 2)

17.1.2 Get Job List

This method retrieves a list of jobs.

Table 778 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
03.01.01		subRC_SUCCESS subRC_APP_JOB_FILTER_INVALID subRC_DB_JOB_ID_LOOKUP_FAILED

Table 778 Get Job List Return Codes

Table 779 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
JobId	The ID of a job in the format <operatorId>_<jobSequenceNumber>.	49	Character string	N/A

Table 779 Get Job List Input Fields

17.1.3 Get Job List By Option

This method retrieves a job list by option.

Table 780 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
03.01.01		subRC_SUCCESS subRC_APP_JOB_FILTER_INVALID subRC_DB_JOB_ID_LOOKUP_FAILED

Table 780 Get Job List By Option Return Codes

Table 781 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default Value
GetJobList OptionList	List of options used for filtering of jobs displayed.	array of enum	FilterByBgName FilterByOperatorId	empty array

Table 781 Get Job List By Option Input Fields

Table 782 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
JobId	The ID of a job in the format <operatorId>_<jobSequenceNumber>.	49	Character string.	N/A

Table 782 Get Job List By Option Output Fields

17.1.4 Cancel Job

This method terminates the processing of the specified currently running or waiting job.

Table 783 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
03.01.01		subRC_SUCCESS subRC_APP_JOB_ID_LEN_INVALID subRC_APP_COMM_OPERATOR_INVALID subRC_APP_JOB_COMMENTS_LEN_INVALID subRC_DB_JOB_ID_LOOKUP_FAILED subRC_APP_JOB_ID_DOES_NOT_EXIST

Table 783 Cancel Job Return Codes

Table 784 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
JobId	The ID of a job in the format <operatorId>_<jobSequenceNumber>.	49	Character string.	N/A
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
JobComments	A text string optionally provided by the operator when a job is cancelled.	2000	Character string.	N/A

Table 784 Cancel Job Input Fields

17.1.5 Generic Job

This method supports submission of generic jobs.

Table 785 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
V7.01.15.01		subRC_SUCCESS others

Table 785 Generic Job Return Codes

Table 786 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorID	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			N/A
JobAction	The action to be performed by this job.	enum	AddFeature RemoveFeature UpdateFeature RemoveCaleaEntry UpdateMlhgMemberCaleaEntry RemoveMlhgMemberCaleaEntry UpdateBGDefaultQOSProfile UpdateQOSProfile AuditFeatureProfile CollectLogFiles GenericAuditHomeDn	N/A

Table 786 Generic Job Input Fields

Table 787 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
JobId	The ID of a job in the format <operatorId>_<jobSequenceNumber>.	49	Character string.	N/A

Table 787 Genric Job Output Fields

18 Dynamic CAC Router Management

The following subsections describe the SOAP/XML interfaces that exist to support Dynamic CAC (Call Admission Control) router operations.

Note: These SOAP requests and responses are separately defined in the interface definition (WSDL) file: siemens-hiq8000.cac.wsdl.

18.1 Notify CAC Primary Link Status

This method reports the status of the primary link on the router.

Table 788 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
11.00.01.25		subRC_SUCCESS subRC_APP_CACSDK_INVALID_CAC_PRIMARYLINK_IPADDRESS subRC_APP_CACSDK_MODIFY_STATUS_ERROR

Table 788 Notify CAC Primary Link Status Return Codes

Table 789 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId'	Please refer to Section 5.4, "Primitive Data Definitions", on page 66.			required
CACPrimaryLinkIpAddress	The IP address of the router for which the link status is to be reported.		A valid IP address	required
CACPrimaryLinkStatus	The link status to report.	enum	LinkUp LinkDown	required

Table 789 Notify CAC Primary Link Status Input Fields

18.2 Get CAC Primary Link Status

This method returns the status of the primary link associated to the router specified by the input CACPrimaryLinkIpAddress or, if CACPrimaryLinkIpAddress is not specified, returns a list of all Access Routers along with their status.

Table 790 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
11.00.01.25		subRC_SUCCESS subRC_APP_CACSDK_INVALID_CAC_PRIMARYLINK_IPADDRESS subRC_SYS_CACSDK_MEMORY_ALLOCATION_ERROR subRC_APP_CACSDK_QUERY_CAC_GROUP_ERROR

Table 790 Get CAC Primary Link Status Return Codes

Table 791 lists the input fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			
InterfaceWSDL	The WSDL version of the request.	Refer to Section 5.1, “Interface (WSDL) Versions”, on page 61.	Character string	
Paging	Paging structure for Get List requests. For more information, see Section 5.5, “Paging”, on page 72.			
CACPrimaryLinkIp Address	The IP address of the router for which the link status list is to be retrieved.		A valid IP address	

Table 791 Get CAC Primary Link Status Input Fields

18.3 Get Provisioned CAC Primary Links

This method searches all the CAC Groups that have a router address configured and returns the address along with the interface name back in a list. The external management application can use this to identify all the routers to query.

Table 792 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
11.00.01.25		subRC_SUCCESS subRC_SYS_CACSDK_MEMORY_ALLOCATION_ERROR subRC_APP_CACSDK_QUERY_CAC_GROUP_ERROR

Table 792 Get Provisioned CAC Primary Links Return Codes

Table 793 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default Value
InterfaceWSDL	The WSDL version of the request.	Refer to Section 5.1, “Interface (WSDL) Versions”, on page 61.	Character string	
Paging	Paging structure for Get List requests. For more information, see Section 5.5, “Paging”, on page 72.			

Table 793 Get Provisioned CAC Primary Links Input Fields

Table 794 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
ProvisionedCACPrimaryLinksList	A list of CACPrimaryLink elements, each containing the following attributes:			
LinkIPAddress	An IP Address string representing the router associated with the CAC Group serving the WAN.	Please refer to the definition of RouterIpAddress in Section 8.10.1, “Create CAC Group”.		N/A
LinkName	This identifies the interface within the router. The string is of the form “eth0”, “eth1” etc.	Please refer to the definition of InterfaceName in Section 8.10.1, “Create CAC Group”.		N/A

Table 794 Get Provisioned CAC Primary Links Output Fields

18.4 Reset All CAC Primary Links

This method resets the status of the primary link for all CAC Access Routers to reflect a status “LinkUp”.

Table 795 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
11.00.01.25		subRC_SUCCESS subRC_APP_CACSDK_QUERY_CAC_GROUP_ERROR subRC_APP_CACSDK_STATUS_PARTIALLY_RESET

Table 795 Reset All CAC Primary Links Return Codes

Table 796 lists the input fields for this method.

Field Name	Description	Length	Range/Value	Default Value
OperatorId	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66.			

Table 796 Reset All CAC Primary Links Input Fields

19 Executive Assistant Functionality with Cockpit Application

The new capabilities and functions that support the requirements of executive assistant groups are outlined in this section. These provisioning and management functions were previously handled by the OpenScape Voice Assistant. Beginning with V4.0, the SOAP interface will be providing the following capabilities:

- Provisioning of E/A Groups
- Provisioning of E/A Group Members
- Provisioning of E/A Member Auxiliary Lines
- Provisioning of E/A Member Relations
- Provisioning of E/A Cockpit Data

19.1 Executive Assistant Groups

The following subsections describe the SOAP/XML interfaces that exist for the provisioning of the executive assistant groups.

19.1.1 Create Executive Assistant Group

This method creates an executive assistant group.

Table 797 lists the return codes for this method.

WSDL Version	Return Codes
V4.00.01.24	subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_EA_BGNAME_INVALID subRC_APP_EA_SERVICE_ID_INVALID subRC_APP_EA_GROUP_FIELD_INVALID subRC_APP_EA_GROUP_EADM_ERROR subRC_APP_EA_ALLOCATION_ERROR subRC_APP_EA_INITIALIZATION_ERROR subRC_DB_EA_ERROR subRC_APP_EA_GROUP_COULD_NOT_BE_REMOVED

Table 797 Create Executive Assistant Group Return Codes (Sheet 1 of 2)

WSDL Version	Return Codes
	subRC_APP_EA_GROUP_NAME_INVALID_LENTGH subRC_APP_MAX_NUMBER_OF_EA_GROUPS_REACHED subRC_APP_EA_GROUP_NAME_MISSING subRC_APP_EA_GROUP_TYPE_MISSING

Table 797 Create Executive Assistant Group Return Codes (Sheet 2 of 2)

Table 798 lists the input fields for this method:

Field Name	Description	Type	Range
Operator ID	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. Required	Str	
BGName	The name of an existing BG in which the Group will be included. Required	Str	
EAGroupName	The name of the new executive assistant group. Required	Str	
EAGroupType	The type of the E/A group.	enum	EAG_Basic, EAG_EnhancedWithCallWaiting, EAG_EnhancedWithRollover, EAG_BasicWithCockpit, EAG_EnhancedWithCallWaitingAndCockpit, EAG_EnhancedWithRolloverAndCockpit
ServerAddress	Server IP of the EA group	Str	
ServerPort	Server Port of the E/A group	Int	
NotificPeerAssist	Flag indicating if notification of peer assistant is required	boolean	
NotificPeerAssistTimeDelay	Value indicating the delay in seconds for the notification of Assistant	Int	
ScreenQueingMode	Flag indicating if a Screen Queuing mode is needed	boolean	

Table 798 Create Executive Assistant Group Input Fields

19.1.2 Delete Executive Assistant Group

This method deletes an executive assistant group.

Table 799 lists the return codes for this method.

WSDL Version	Return Codes
V4.00.01.24	subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID

Table 799 Delete Executive Assistant Group Return Codes (Sheet 1 of 2)

WSDL Version	Return Codes
	subRC_APP_BG_NAME_LEN_INVALID
	subRC_APP_EA_BGNAME_INVALID
	subRC_APP_EA_SERVICE_ID_INVALID
	subRC_APP_EA_GROUP_FIELD_INVALID
	subRC_APP_EA_GROUP_EADM_ERROR
	subRC_APP_EA_ALLOCATION_ERROR
	subRC_APP_EA_INITIALIZATION_ERROR
	subRC_DB_EA_ERROR
	subRC_APP_EA_GROUP_COULD_NOT_BE_REMOVED
	subRC_APP_EA_GROUP_NOT_FOUND
	subRC_APP_EA_GROUP_RETRIEVAL_FAILED

Table 799 Delete Executive Assistant Group Return Codes (Sheet 2 of 2)

Table 800 lists the input fields for this method:

Field Name	Description	Type	Range
Operator ID	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. Required	Str	
BGName	The name of an existing BG in which the EA Group is included. Required	Str	
EAGroupName	The name of the executive assistant group to be deleted. Required	Str	

Table 800 Delete Executive Assistant Group Input Fields

19.1.3 Update Executive Assistant Group

This method updates an existing executive assistant group. If the group includes members, this method allows the modification of corresponding member information. The return codes for this function are:

Table 801 lists the return codes for this method.

WSDL Version	Return Codes
V4.00.01.24	subRC_SUCCESS
	subRC_APP_COMM_OPERATOR_INVALID
	subRC_APP_BG_NAME_LEN_INVALID
	subRC_APP_EA_BGNAME_INVALID
	subRC_APP_EA_SERVICE_ID_INVALID
	subRC_APP_EA_GROUP_FIELD_INVALID
	subRC_APP_EA_GROUP_EADM_ERROR

Table 801 Update Executive Assistant Group Return Codes (Sheet 1 of 2)

WSDL Version	Return Codes
	subRC_APP_EA_ALLOCATION_ERROR
	subRC_APP_EA_INITIALIZATION_ERROR
	subRC_DB_EA_ERROR
	subRC_APP_EA_GROUP_NOT_FOUND
	subRC_APP_EA_GROUP_RETRIEVAL_FAILED
	subRC_APP_EA_GROUP_NAME_MISSING
	subRC_APP_EA_GROUP_TYPE_MISSING

Table 801 Update Executive Assistant Group Return Codes (Sheet 2 of 2)

Table 802 lists the input fields for this method:

Field Name	Description	Type	Range
Operator ID	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. Required	Str	
BGName	The name of an existing BG in which the EA Group is included. Required	Str	
EAGroupName	The name of the executive assistant group to be updated. Required	Str	
NewEAGroupName	The new name of the executive assistant group to be updated. Required	Str	optional
EAGroupType	The type of the E/A group.	Enum	EAG_Basic, EAG_EnhancedWithCallWaiting, EAG_EnhancedWithRollover, EAG_BasicWithCockpit, EAG_EnhancedWithCallWaitingAndCockpit, EAG_EnhancedWithRolloverAndCockpit
ServerAddress	Server IP of the EA group	Str	
ServerPort	Server Port of the E/A group	Int	
NotificPeerAssist	Flag indicating if notification of peer assistant is required	boolean	
NotificPeerAssistTimeDelay	Value indicating the delay in seconds for the notification of Assistant	Int	
ScreenQueingMode	Flag indicating if a Screen Queuing mode is needed	boolean	

Table 802 Update Executive Assistant Group Input Fields

Optionally the craft person could provide a list of members that should be updated as well. Each one would consist of the following attributes:

- MemberServiceID
- MemberRole
- PrivateNumberVisible

- MemberName
- TypeExecutive
- MemberPreference
- MemberMobileNumber
- MemberForwardNumber
- MemberToNumber
- AdditionalInfo

The optional member attributes are described in Section 19.2.1, “Create Executive Assistant Group Member”, on page 674.

19.1.4 Get List of Executive Assistant Groups

This method can be used in order to get a list of all EA Groups within a specific BG or to get the information of a single EA Group.

Table 804 lists the return codes for this method.

WSDL Version	Return Codes
V4.00.01.24	subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_EA_BGNAME_INVALID subRC_APP_EA_GROUP_EADM_ERROR subRC_APP_EA_ALLOCATION_ERROR subRC_APP_EA_INITIALIZATION_ERROR subRC_DB_EA_ERROR subRC_APP_EA_GROUP_RETRIEVAL_FAILED
V5.00.20.01	subRC_APP_OSMO_DN_CONFLICT

Table 803 Get List of Executive Assistant Group Return Codes

Table 806 lists the input fields for this method:

Field Name	Description	Type	Range
Operator ID	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. Required	Str	
BGName	The name of an existing BG to use as a filter. Required	Str	
EAGroupName	The name of the executive assistant group to look for.	Str	

Table 804 Get List of Executive Assistant Group Input Fields

Field Name	Description	Type	Range
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.		
Sort Order	Value indicating the sort order of the list	Enum	SortOrderAscending, SortOrderDescending
Sort Type	Value indicating the sort type of the list	Enum	SortByGroupID, SortByGroupName

Table 804 Get List of Executive Assistant Group Input Fields

19.1.5 Get Extended Executive Assistant Group List

GetEAGroupList() is used to retrieve the list of EA Groups within a specific BG. Since that list does not include the Number of Executives, the Number of Assistants and the Cockpit Status that the Assistant needs for its list, a new API has been provided, getEAGroupListExtended(), that will return the needed data for the EA Group List. That includes: EA Group Name, EA Group Type, the Number of Executives, the Number of Assistants and the Cockpit Status. Paging is provided and sorting/filtering based on Group Name. EADM provides sorting/filtering/paging facilities plus the full result-set to SOAP. Input Fields are the same as Get List of Executive Assistant Groups described in section 19.1.4

The following table shows the Extended Result parameters

Table 806 lists the output fields for this method:

Field Name	Description	Type	Range
BgName	The name of an existing BG in which the EA Group is included. Required		
EAGroupName	The name of the executive assistant group	Str	
EAGroupType	The type of the E/A group.	Enum	EAG_Basic, EAG_EnhancedWithCallWaiting, EAG_EnhancedWithRollover, EAG_BasicWithCockpit, EAG_EnhancedWithCallWaitingAndCockpit, EAG_EnhancedWithRolloverAndCockpit
ExecutiveCount	Count of Executives	Integer	
Assistant Count	Count of Assistants	integer	

Table 805 Get Extended Executive Assistant Group List Output Fields

Field Name	Description	Type	Range
Cockpit Status	Whether it Cockpit Application or not	Boolean	

Table 805 Get Extended Executive Assistant Group List Output Fields

19.2 Executive Assistant Group Members

The following subsections describe the SOAP/XML interfaces that exist for the provisioning of the executive assistant group Members.

19.2.1 Create Executive Assistant Group Member

This method creates an executive assistant group member.

Table 806 lists the return codes for this method.

WSDL Version	Return Codes
V4.00.01.24	subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_SUB_DN_INVALID subRC_APP_SUB_DN_DOES_NOT_EXIST subRC_DB_SUB_DN_LOOKUP_FAILED subRC_APP_EA_BGNAME_INVALID subRC_APP_EA_SERVICE_ID_INVALID subRC_APP_EA_GROUP_MEMBER_FIELD_INVALID subRC_APP_EA_GROUP_MEMBER_EADM_ERROR subRC_APP_EA_ALLOCATION_ERROR subRC_APP_EA_INITIALIZATION_ERROR subRC_DB_EA_ERROR subRC_APP_EA_COCKPIT_INFO_NOT_SPECIFIED subRC_APP_EA_GROUP_MEMBERS_EXIST subRC_APP_EA_GROUP_NOT_FOUND subRC_APP_EA_MAX_EA_MEMBER_NUMBER_PER_GROUP subRC_APP_EA_GROUP_RETRIEVAL_FAILED subRC_APP_EA_MEMBER_EXISTS subRC_APP_EA_MEMBER_NOT_FOUND

Table 806 Create Executive Assistant Group Member Return Codes

Table 807 lists the input fields for this method:

Field Name	Description	Type	Range
Operator ID	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. Required	Str	
BGName	The name of an existing BG in which the Group will be included. Required	Str	
EAGroupName	The name of an existing group to which the member will be included. Required	Str	
MemberServiceID	The ServiceID / DN of the new member	Str	
MemberRole	A value indicating the role of the Member	Enum	EAM_Executive, EAM_Secretary, EAM_Deputy
PrivateNumberVisible	Flag indicating if the Private number of the member would be visible	Boolean	
MemberName	A name that can be used for the member	Str	
TypeExecutive	Flag indicating if member is an executive	Boolean	
MemberPreference	Attribute that can be used for preferences	Str	
MemberMobileNumber	A value indicating the mobile number of the member	Str	
MemberForwardNumber	A value indicating the forward number of the member	Str	
MemberToNumber	A value indicating the 'to' number of the member	Str	
AdditionalInfo	Additional information	Str	

Table 807 Create Executive Assistant Group Member Input Fields

19.3 Delete Executive Assistant Group Member

This method deletes an executive assistant group member.

Table 808 lists the return codes for this method.

WSDL Version	Return Codes
V4.00.01.24	subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_SUB_DN_INVALID subRC_APP_SUB_DN_DOES_NOT_EXIST subRC_DB_SUB_DN_LOOKUP_FAILED subRC_APP_EA_BGNAME_INVALID subRC_APP_EA_SERVICE_ID_INVALID
	subRC_APP_EA_GROUP_MEMBER_FIELD_INVALID subRC_APP_EA_GROUP_MEMBER_EADM_ERROR

Table 808 Delete Executive Assistant Group Member Return Codes

WSDL Version	Return Codes
	subRC_APP_EA_MEMBER_COULD_NOT_BE_REMOVED
	subRC_APP_EA_ALLOCATION_ERROR
	subRC_APP_EA_INITIALIZATION_ERROR
	subRC_DB_EA_ERROR
	subRC_APP_EA_GROUP_NOT_FOUND
	subRC_APP_EA_GROUP_RETRIEVAL_FAILED
	subRC_APP_EA_MEMBER_NOT_FOUND

Table 808 Delete Executive Assistant Group Member Return Codes

Table 809 lists the input fields for this method:

Field Name	Description	Type	Range
Operator ID	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. Required	Str	
BGName	The name of an existing BG in which the Group is included. Required	Str	
EAGroupName	The name of an existing group containing the member to be deleted. Required	Str	
MemberServiceID	The ServiceID / DN of the member to be deleted. Required	Str	

Table 809 Delete Executive Assistant Group Member Input Fields

19.3.1 Get List of Executive Assistant Group Members

This method can be used in order to get either a list of all members of a specific EA Group or to get the information of a single member.

Table 810 lists the return codes for this method.

WSDL Version	Return Codes
V4.00.01.24	subRC_SUCCESS
	subRC_APP_COMM_OPERATOR_INVALID
	subRC_APP_BG_NAME_LEN_INVALID
	subRC_APP_SUB_DN_INVALID
	subRC_APP_SUB_DN_DOES_NOT_EXIST
	subRC_DB_SUB_DN_LOOKUP_FAILED
	subRC_APP_EA_BGNAME_INVALID
	subRC_APP_EA_SERVICE_ID_INVALID
	subRC_APP_EA_GROUP_MEMBER_FIELD_INVALID

Table 810 Get List of Executive Assistant Group Members Return Codes

WSDL Version	Return Codes
	subRC_APP_EA_GROUP_MEMBER_EADM_ERROR
	subRC_APP_EA_ALLOCATION_ERROR
	subRC_APP_EA_INITIALIZATION_ERROR
	subRC_DB_EA_ERROR
	subRC_APP_EA_GROUP_NOT_FOUND
	subRC_APP_EA_GROUP_RETRIEVAL_FAILED
	subRC_APP_EA_MEMBER_NOT_FOUND

Table 810 Get List of Executive Assistant Group Members Return Codes

Table 811 lists the input fields for this method:

Field Name	Description	Type	Range
Operator ID	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. Required	Str	
BGName	The name of an existing BG in which the Group is included. Required	Str	
EAGroupName	The name of an existing group containing the member to be deleted. Required	Str	
MemberServiceID	The ServiceID / DN of the member to be deleted	Str	
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.		
Sort Order	Value indicating the sort order of the list	Enum	SortOrderAscending, SortOrderDescending

Table 811 Get List of Executive Assistant Group Members Input Fields

19.4 Executive Assistant Group Member Relations

The following subsections describe the SOAP/XML interfaces that exist for the provisioning of Relations between executive assistant group members.

19.4.1 Create Executive Assistant Group Member Relation

This method creates a relation between 2 existing executive assistant group members.

Table 812 lists the return codes for this method.

WSDL Version	Return Codes
V4.00.01.24	subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_SUB_DN_INVALID subRC_APP_SUB_DN_DOES_NOT_EXIST subRC_DB_SUB_DN_LOOKUP_FAILED subRC_APP_EA_BGNAME_INVALID subRC_APP_EA_SERVICE_ID_INVALID subRC_APP_EA_GROUP_RELATION_FIELD_INVALID subRC_APP_EA_GROUP_RELATION_EADM_ERROR subRC_APP_EA_ALLOCATION_ERROR subRC_APP_EA_INITIALIZATION_ERROR subRC_DB_EA_ERROR subRC_APP_EA_SERVICEID_IS_USED_AS_EA_MEMBER subRC_APP_EA_GROUP_RETRIEVAL_FAILED subRC_APP_EA_MEMBER_NOT_FOUND subRC_APP_EA_RELATION_MISMATCH

Table 812 Create Executive Assistant Group Member Relation Return Codes

Table 813 lists the input fields for this method:

Field Name	Description	Type	Range
Operator ID	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. Required	Str	
From Member	The Service ID of the 'from member’. Required	'Str	
To Member	The Service ID of the 'to member'. Required	Str	
Relation Order	The type of the relation.	enum	EAR_ExecutiveToPrimaryAssistant, EAR_ExecutiveToSecondaryAssistant, EAR_AssistantToDeputy

Table 813 Create Executive Assistant Group Member Relation Input Fields

19.4.2 Delete Executive Assistant Group Member Relation

This method removes an existing relation between 2 executive assistant group members.

Table 814 lists the return codes for this method.

WSDL Version	Return Codes
V4.00.01.24	subRC_SUCCESS
	subRC_APP_COMM_OPERATOR_INVALID
	subRC_APP_BG_NAME_LEN_INVALID
	subRC_APP_SUB_DN_INVALID
	subRC_APP_SUB_DN_DOES_NOT_EXIST
	subRC_DB_SUB_DN_LOOKUP_FAILED
	subRC_APP_EA_BGNAME_INVALID
	subRC_APP_EA_SERVICE_ID_INVALID
	subRC_APP_EA_GROUP_RELATION_FIELD_INVALID
	subRC_APP_EA_GROUP_RELATION_EADM_ERROR
	subRC_APP_EA_ALLOCATION_ERROR
	subRC_APP_EA_INITIALIZATION_ERROR
	subRC_DB_EA_ERROR
	subRC_APP_EA_GROUP_RETRIEVAL_FAILED
	subRC_APP_EA_MEMBER_NOT_FOUND
	subRC_APP_EA_RELATION_REMOVAL_FAILED

Table 814 Delete Executive Assistant Group Member Relation Return Codes

Table 815 lists the input fields for this method:

Field Name	Description	Type	Range
Operator ID	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. Required	Str	
From Member	The Service ID of the 'from member. Required	'Str	
To Member	The Service ID of the 'to member'. Required	Str	
Relation Order	The type of the relation.	enum	EAR_ExecutiveToPrimaryAssistant, EAR_ExecutiveToSecondaryAssistant, EAR_AssistantToDeputy

Table 815 Delete Executive Assistant Group Member Relation Input Fields

19.4.3 Get Executive Assistant Group Member Relation List

This method can be used in order to get either a list of relations all members of a specific EA Group.

Table 816 lists the return codes for this method.

WSDL Version	Return Codes
V4.00.01.24	subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_SUB_DN_INVALID subRC_APP_SUB_DN_DOES_NOT_EXIST subRC_DB_SUB_DN_LOOKUP_FAILED subRC_APP_EA_BGNAME_INVALID subRC_APP_EA_SERVICE_ID_INVALID subRC_APP_EA_GROUP_RELATION_FIELD_INVALID subRC_APP_EA_GROUP_RELATION_EADM_ERROR subRC_APP_EA_ALLOCATION_ERROR subRC_APP_EA_INITIALIZATION_ERROR subRC_DB_EA_ERROR

Table 816 Get Executive Assistant Group Member Relation Return Codes

Table 817 lists the input fields for this method:

Field Name	Description	Type	Range
Operator ID	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. Required	Str	
BGName	The name of an existing BG in which the Group is included. Required	Str	
EAGroupName	The name of an existing group containing the member to be filtered. Required	Str	
MemberServiceID	The ServiceID / DN to be filtered	Str	
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.		
Sort Order	Value indicating the sort order of the list	Enum	SortOrderAscending, SortOrderDescending

Table 817 Get Executive Assistant Group Member Relation Input Fields

19.5 Executive Assistant Group Member Auxiliary Lines

The following subsections describe the SOAP/XML interfaces that exist for the provisioning of Auxiliary Lines for executive assistant group Members.

19.5.1 Create Executive Assistant Group Member Auxiliary Line

This method creates an auxiliary line for an existing executive assistant group member.

Table 818 lists the return codes for this method.

WSDL Version	Return Codes
V4.00.01.24	subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_SUB_DN_INVALID subRC_APP_SUB_DN_DOES_NOT_EXIST subRC_DB_SUB_DN_LOOKUP_FAILED subRC_APP_EA_BGNAME_INVALID subRC_APP_EA_SERVICE_ID_INVALID subRC_APP_EA_GROUP_MEMBER_AUX_LINE_FIELD_INVALID subRC_APP_EA_GROUP_MEMBER_AUX_LINE_EADM_ERROR subRC_APP_EA_ALLOCATION_ERROR subRC_APP_EA_INITIALIZATION_ERROR subRC_DB_EA_ERROR subRC_APP_EA_GROUP_NOT_FOUND subRC_APP_EA_GROUP_RETRIEVAL_FAILED subRC_APP_MAX_AUX_LINES_PER_MEMBER_REACHED subRC_APP_MAX_AUX_LINES_REACHED subRC_APP_EA_GROUP_AUX_LINES_ALREADY_CREATED subRC_APP_EA_GROUP_AUX_LINE_NOT_FOUND

Table 818 Create Executive Assistant Group Member Auxiliary Line Return Codes

Table 819 lists the input fields for this method:

Field Name	Description	Type	Range
Operator ID	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. Required	Str	
Member Service ID	The Service ID of the member to which the aux line will be assigned. Required	Str	
Auxiliary line Service ID	The Service ID of the auxiliary line. Required	Str	
Auxiliary line type	The type of the auxiliary line.	enum	EAX_Overflow, EAX_Intercom, EAX_Private

Table 819 Create Executive Assistant Group Member Auxiliary Line Input Fields

19.5.2 Delete Executive Assistant Group Member Auxiliary Line

This method deletes an existing auxiliary line for an executive assistant group member.

Table 820 lists the return codes for this method.

WSDL Version	Return Codes
V4.00.01.24	subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_SUB_DN_INVALID subRC_APP_SUB_DN_DOES_NOT_EXIST subRC_DB_SUB_DN_LOOKUP_FAILED subRC_APP_EA_BGNAME_INVALID subRC_APP_EA_SERVICE_ID_INVALID subRC_APP_EA_GROUP_MEMBER_AUX_LINE_FIELD_INVALID subRC_APP_EA_GROUP_MEMBER_AUX_LINE_EADM_ERROR subRC_APP_EA_ALLOCATION_ERROR subRC_APP_EA_INITIALIZATION_ERROR subRC_DB_EA_ERROR subRC_APP_EA_GROUP_NOT_FOUND subRC_APP_EA_GROUP_RETRIEVAL_FAILED subRC_APP_EA_GROUP_AUX_LINE_NOT_FOUND

Table 820 Delete Executive Assistant Group Member Auxiliary Line Return Codes

Table 821 lists the input fields for this method:

Field Name	Description	Type	Range
Operator ID	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. Required	Str	
Member Service ID	The Service ID of the member of which the aux line will be assigned. Required	Str	
Auxiliary line Service ID	The Service ID of the auxiliary line. Required	Str	

Table 821 Delete Executive Assistant Group Member Auxiliary Line Input Fields

19.5.3 Get Executive Assistant Group Member Relations

This method can be used in order to get a list of all Auxiliary Lines of a specified Member of a certain EA Group, or a list of Auxiliary Lines of all Members within a specified EA Group.

Table 822 lists the return codes for this method.

WSDL Version	Return Codes
V4.00.01.24	subRC_SUCCESS
	subRC_APP_COMM_OPERATOR_INVALID
	subRC_APP_BG_NAME_LEN_INVALID
	subRC_APP_SUB_DN_INVALID
	subRC_APP_SUB_DN_DOES_NOT_EXIST
	subRC_DB_SUB_DN_LOOKUP_FAILED
	subRC_APP_EA_BGNAME_INVALID
	subRC_APP_EA_SERVICE_ID_INVALID
	subRC_APP_EA_GROUP_MEMBER_AUX_LINE_FIELD_INVALID
	subRC_APP_EA_GROUP_MEMBER_AUX_LINE_EADM_ERROR
	subRC_APP_EA_ALLOCATION_ERROR
	subRC_APP_EA_INITIALIZATION_ERROR
	subRC_DB_EA_ERROR
	subRC_APP_EA_GROUP_NOT_FOUND
	subRC_APP_EA_GROUP_RETRIEVAL_FAILED

Table 822 Get Executive Assistant Group Member Auxiliary Line Return Codes

Table 823 lists the input fields for this method:

Field Name	Description	Type	Range
Operator ID	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. Required	Str	
MemberServiceID	The ServiceID / DN to be filtered	Str	
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.		
Sort Order	Value indicating the sort order of the list	Enum	SortOrderAscending, SortOrderDescending

Table 823 Get Executive Assistant Group Member Auxiliary Line Input Fields

19.6 Executive Assistant Cockpit

The following subsections describe the SOAP/XML interfaces that exist for the provisioning of cockpit information regarding for executive assistant group members.

19.6.1 Get Executive Assistant Cockpit List

This method can be used in order to get a list of all EA Groups within a specific Business Group that are of type 'Cockpit'.

Table 824 lists the return codes for this method.

WSDL Version	Return Codes
V4.00.01.24	subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_SUB_DN_INVALID subRC_APP_SUB_DN_DOES_NOT_EXIST subRC_DB_SUB_DN_LOOKUP_FAILED subRC_APP_EA_BGNAME_INVALID subRC_APP_EA_GROUP_EADM_ERROR subRC_APP_EA_ALLOCATION_ERROR subRC_APP_EA_INITIALIZATION_ERROR subRC_DB_EA_ERROR

Table 824 Get Executive Assistant Cockpit List Return Codes

Table 825 lists the input fields for this method:

Field Name	Description	Type	Range
Operator ID	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. Required	Str	
BGName	The name of an existing BG in which the Group is included. Required	Str	
Paging	Paging structure for Get List requests. For more information, refer to Section 5.5, “Paging”, on page 72.		
Sort Order	Value indicating the sort order of the list	Enum	SortOrderAscending, SortOrderDescending

Table 825 Get Executive Assistant Cockpit List Input Fields

19.6.2 Get Executive Assistant Cockpit Configuration Data

This method can be used in order to get the configuration data of all Members of a certain E/A Group/Cockpit.

Table 826 lists the return codes for this method.

WSDL Version	Return Codes
V4.00.01.24	subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_BG_NAME_LEN_INVALID subRC_APP_SUB_DN_INVALID subRC_APP_SUB_DN_DOES_NOT_EXIST subRC_DB_SUB_DN_LOOKUP_FAILED subRC_APP_EA_BGNAME_INVALID subRC_APP_EA_SERVICE_ID_INVALID subRC_APP_EA_GROUP_FIELD_INVALID subRC_APP_EA_GROUP_MEMBER_FIELD_INVALID subRC_APP_EA_GROUP_RELATION_FIELD_INVALID subRC_APP_EA_GROUP_MEMBER_AUX_LINE_FIELD_INVALID subRC_APP_EA_GROUP_COCKPIT_FIELD_INVALID subRC_APP_EA_GROUP_EADM_ERROR subRC_APP_EA_GROUP_MEMBER_EADM_ERROR subRC_APP_EA_GROUP_RELATION_EADM_ERROR subRC_APP_EA_GROUP_MEMBER_AUX_LINE_EADM_ERROR subRC_APP_EA_GROUP_COCKPIT_EADM_ERROR subRC_APP_EA_ALLOCATION_ERROR subRC_APP_EA_INITIALIZATION_ERROR subRC_DB_EA_ERROR subRC_APP_EA_GROUP_NOT_FOUND subRC_APP_EA_GROUP_RETRIEVAL_FAILED subRC_APP_EA_GROUP_NAME_MISSING subRC_APP_EA_GROUP_TYPE_MISSING

Table 826 Get Executive Assistant Cockpit Configuration Data Return Codes

Table 827 lists the input fields for this method:

Field Name	Description	Type	Range
Operator ID	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. Required	Str	

Table 827 Get Executive Assistant Cockpit Configuration Data Input Fields

Field Name	Description	Type	Range
BGName	The name of an existing BG in which the Group is included. Required	Str	
Executive Assistant Group Name	The E/A group name that will be used as a filter. Required	Str	

Table 827 Get Executive Assistant Cockpit Configuration Data Input Fields

19.6.3 Get Executive Assistant Cockpit Dynamic Data

This method can be used in order to get the dynamic data of all members of a certain E/A Group/Cockpit.

Table 828 lists the return codes for this method.

WSDL Version	Return Codes
V4.00.01.24	subRC_SUCCESS
	subRC_APP_COMM_OPERATOR_INVALID
	subRC_APP_BG_NAME_LEN_INVALID
	subRC_APP_SUB_DN_INVALID
	subRC_APP_SUB_DN_DOES_NOT_EXIST
	subRC_DB_SUB_DN_LOOKUP_FAILED
	subRC_APP_EA_BGNAME_INVALID
	subRC_APP_EA_SERVICE_ID_INVALID
	subRC_APP_EA_GROUP_FIELD_INVALID
	subRC_APP_EA_GROUP_MEMBER_FIELD_INVALID
	subRC_APP_EA_GROUP_RELATION_FIELD_INVALID
	subRC_APP_EA_GROUP_MEMBER_AUX_LINE_FIELD_INVALID
	subRC_APP_EA_GROUP_COCKPIT_FIELD_INVALID
	subRC_APP_EA_GROUP_EADM_ERROR
	subRC_APP_EA_GROUP_MEMBER_EADM_ERROR
	subRC_APP_EA_GROUP_RELATION_EADM_ERROR
	subRC_APP_EA_GROUP_MEMBER_AUX_LINE_EADM_ERROR
	subRC_APP_EA_GROUP_COCKPIT_EADM_ERROR
	subRC_APP_EA_ALLOCATION_ERROR
	subRC_APP_EA_INITIALIZATION_ERROR
	subRC_DB_EA_ERROR
	subRC_APP_EA_GROUP_NOT_FOUND
	subRC_APP_EA_GROUP_RETRIEVAL_FAILED
	subRC_APP_EA_GROUP_NAME_MISSING

Table 828 Get Executive Assistant Cockpit Dynamic Data Return Codes

WSDL Version	Return Codes
	subRC_APP_EA_GROUP_TYPE_MISSING

Table 828 Get Executive Assistant Cockpit Dynamic Data Return Codes

Table 829 lists the input fields for this method:

Field Name	Description	Type	Range
Operator ID	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. Required	Str	
BGName	The name of an existing BG in which the Group is included. Required	Str	
EAGroupName	The E/A group name that will be used as a filter. Required	Str	

Table 829 Get Executive Assistant Cockpit Dynamic Data Input Fields

19.6.4 Update Executive Assistant Cockpit Dynamic Data

This method can be used in update the dynamic data of a specific members of a certain E/A Group/Cockpit.

Table 830 lists the return codes for this method.

WSDL Version	Return Codes
V4.00.01.24	subRC_SUCCESS
	subRC_APP_COMM_OPERATOR_INVALID
	subRC_APP_BG_NAME_LEN_INVALID
	subRC_APP_SUB_DN_INVALID
	subRC_APP_SUB_DN_DOES_NOT_EXIST
	subRC_DB_SUB_DN_LOOKUP_FAILED
	subRC_APP_EA_BGNAME_INVALID
	subRC_APP_EA_SERVICE_ID_INVALID
	subRC_APP_EA_GROUP_FIELD_INVALID
	subRC_APP_EA_GROUP_MEMBER_FIELD_INVALID
	subRC_APP_EA_GROUP_RELATION_FIELD_INVALID
	subRC_APP_EA_GROUP_MEMBER_AUX_LINE_FIELD_INVALID
	subRC_APP_EA_GROUP_COCKPIT_FIELD_INVALID
	subRC_APP_EA_GROUP_EADM_ERROR
	subRC_APP_EA_GROUP_MEMBER_EADM_ERROR
	subRC_APP_EA_GROUP_RELATION_EADM_ERROR

Table 830 Update Executive Assistant Cockpit Dynamic Data Return Codes

WSDL Version	Return Codes
	subRC_APP_EA_GROUP_MEMBER_AUX_LINE_EADM_ERROR subRC_APP_EA_GROUP_COCKPIT_EADM_ERROR subRC_APP_EA_ALLOCATION_ERROR subRC_APP_EA_INITIALIZATION_ERROR subRC_DB_EA_ERROR subRC_APP_EA_GROUP_NOT_FOUND subRC_APP_EA_GROUP_RETRIEVAL_FAILED subRC_APP_EA_GROUP_NAME_MISSING subRC_APP_EA_GROUP_TYPE_MISSING

Table 830 Update Executive Assistant Cockpit Dynamic Data Return Codes

Table 831 lists the input fields for this method:

Field Name	Description	Type	Range
Operator ID	Please refer to Section 5.4, “Primitive Data Definitions”, on page 66. Required	Str	
BGName	The name of an existing BG in which the Group is included. Required	Str	
EAGroupName	The E/A group name that will be used as a filter. Required	Str	
Server Port	A value representing the port of the Server used with the group	Str	
MemberIdentifier	The ServiceID / DN of the member to be updated	Str	
PhoneIP	A value indicating the Phone IP of the member	Enum	EAM_Executive, EAM_Secretary, EAM_Deputy
TypeExecutive	Flag indicating if member is an executive	Boolean	
MemberName	A name that can be used for the member	Str	
MemberStatus	The current status of the EA member	Enum	EAC_EXEC_TO_ASNT_SOFT, EAC_EXEC_TO_ASNT_REG, EAC_EXEC_TO_ASNT_STRICT EAC_EXEC_IMM_RING, EAC_EXEC_TO_EXEC_MOB, EAC_EXEC_TO_EXEC_VMAIL, EAC_EXEC_TO_NUMBER, EAC_ASNT_AT_DESK, EAC_ASNT_OFF_DESK, EAC_ASNT_TO_ASNT_VMAIL, EAC_ASNT_TO_ASNT_MOB, EAC_ASNT_TO_NUMBER
MemberPreference	Attribute that can be used for preferences	Str	
MemberForwardNumber	A value indicating the forward number of the member	Str	

Table 831 Update Executive Assistant Cockpit Dynamic Data Input Fields

Field Name	Description	Type	Range
MemberToNumber	A value indicating the 'to' number of the member	Str	
AdditionalInfo	Additional information	Str	

Table 831 Update Executive Assistant Cockpit Dynamic Data Input Fields

20 Upgrade and Remote Patching

The following subsections describe the SOAP/XML interfaces that exist to support Upgrade and related Remote Patching operations.

Remote Patching capability allows semi-automating of the upgrade process and allows a GUI interface to initiate this action. The role of the SOAP Server for these requests is merely as a transport layer. The requests are forwarded to the Upgrade Library for processing. SOAP will not validate the requests or the sequence of operations attempted, but instead will simply do a “flow through” to the Upgrade Library. Any error message returned will be mapped back and returned to the client.

Note: These SOAP requests and responses are separately defined in the interface definition (WSDL) file: siemens-hiq8000.upgrade.wsdl

---

20.1 Get Node Info

This method retrieves node software version, patch level information, and other data about the system.

Table 832 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
11.00.06.08		subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_SYS_SPT_ALLOC_ERROR subRC_APP_SPT_NO_INIT_ERROR subRC_APP_SPT_GEN_ERROR subRC_SYS_SPT_ALLOC_ERROR subRC_APP_SPT_SERVER_ERROR

Table 832 Get Node Info Return Codes

Table 833 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to the definition of OperatorId in Section 5.4, “Primitive Data Definitions”, on page 66.			required

Table 833 Get Node Info Input Fields

Table 834 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
CEName	The CE (Cluster Element) node name.	1..1024	Character String	N/A
CEIP	The CE node IP address.	1..1024	Character String	N/A
NodeDBRole	Indicates whether the database on the node is currently Primary or Secondary.	enum	RolePrimary RoleSecondary	N/A
NodeId	Indicates whether the node is designated “Node 1” or “Node 2”.	enum	NodeId1 NodeId2	N/A
rtpVersion	The RTP (Resilient Telco Platform) software version currently running on the node.	1..1024	Character String	N/A
patchVersion	The patch set version currently running on the node.	1..1024	Character String	N/A
osType	The operating system type.	1..1024	Character String	N/A
osVersion	The operating system version.	1..1024	Character String	N/A
VIP	Virtual IP address of the cluster.	1..1024	Character String	N/A
dbVersion	The version of the Solid database.	1..1024	Character String	N/A
thirdPartyVersion	The version of third party software on the node.	1..1024	Character String	N/A

Table 834 Get Node Info Output Fields

20.2 Get Node Patch Set List

This method retrieves the list of all patch sets available on a node or on a DVD attached to a node. In response to this operation, a list of patch sets is returned; for example:

```
11.00.01.ALL.12_PS0001 (UNSPps0001)
11.00.01.ALL.12_PS0002 (UNSPps0002)
11.00.01.ALL.12_PS0003 (UNSPps0003)
```

Table 835 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
11.00.06.08		subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_SPT_PATCH_LOCATION_MISSING subRC_APP_SPT_NODE_ID_MISSING subRC_APP_SPT_TIMEOUT_ERROR subRC_APP_SPT_NO_INIT_ERROR subRC_APP_SPT_SERVER_ERROR subRC_SYS_SPT_ALLOC_ERROR

Table 835 Get Node Patch Set List Return Codes

Table 836 lists the *input* fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	Please refer to the definition of OperatorId in Section 5.4, “Primitive Data Definitions”, on page 66.			required
NodeId	Indicates whether the request is for “Node 1” or “Node 2”.	enum	NodeId1 NodeId2	required
PatchLocation	The location of the patch sets to report. Note that the options are OpenScape Voice or DVD; for SIS, the request GetSisPatchSetList must be used.	enum	PatchLocationHIQ PatchLocationDVD	required

Table 836 Get Node Patch Set List Input Fields

Table 837 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
PatchSetList	A list of PatchSet elements, each containing the following attribute:			
PatchSet. PatchSetName	The name of a patch set.	1..1024	Character String	N/A

Table 837 Get Node Patch Set List Output Fields

20.3 Get SIS Patch Set List

This method retrieves the list of all patch sets available on a SIS (Software Installation Server).

Table 838 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
11.00.06.08		subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_SPT_SIS_IP_MISSING subRC_APP_SPT_SIS_USER_MISSING subRC_APP_SPT_SIS_PASSWORD_MISSING subRC_APP_SPT_TIMEOUT_ERROR subRC_APP_SPT_NO_INIT_ERROR subRC_APP_SPT_SERVER_ERROR subRC_SYS_SPT_ALLOC_ERROR

Table 838 Get Sis Patch Set List Return Codes

Table 839 lists the *input* fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	Please refer to the definition of OperatorId in Section 5.4, “Primitive Data Definitions”, on page 66.			required
sisIP	The IP address of the SIS (Software Installation Server).	1..1024	Character String	required
sisUser	The user name with which to log on to the SIS (Software Installation Server).	1..1024	Character String	required
sisIP	The password with which to log on to the SIS (Software Installation Server).	1..1024	Character String	required

Table 839 Get Sis Patch Set List Input Fields

Table 840 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
PatchSetList	A list of PatchSet elements, each containing the following attribute:			
PatchSet. PatchSetName	The name of a patch set.	1..1024	Character String	N/A

Table 840 Get Sis Patch Set List Output Fields

20.4 Upgrade

This method initiates a new upgrade on a system. The upgrade will be performed on the specified node of the system only.

Table 841 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
11.00.06.08		subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_SPT_CLIENT_ERROR subRC_APP_SPT_TIMEOUT_ERROR subRC_APP_SPT_NO_INIT_ERROR subRC_APP_SPT_SERVER_ERROR subRC_SYS_SPT_ALLOC_ERROR subRC_APP_SPT_GEN_ERROR

Table 841 Upgrade Return Codes

Table 842 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to the definition of OperatorId in Section 5.4, “Primitive Data Definitions”, on page 66.			required
NodeId	Indicates whether the request is for “Node 1” or “Node 2”. Note: This operation is expected to be executed on Node 1 only; it is not supported for Node 2. To upgrade the second node, the UpgradeSecondNode operation must be used.	enum	NodeId1 NodeId2	required
upgradeMethod	This specifies whether to use “Online”, ”Rolling” or “Split Mode Upgrade”. (The difference is if RTP uses Online is stores copies of the patches for rollback, while Rolling does not). Note: Only UpgradeMethodRolling is supported at this time.	enum	UpgradeMethodOnline UpgradeMethodRolling UpgradeMethodSMU	UpgradeMethod Rolling
patchSetToInstall	The list of patchsets to be installed. Each element of the list consists of the following attribute:			
PatchSet. PatchSetName	The name of a patch set.	1..1024	Character String	N/A
patchSetToRemove	The list of patchsets to be removed. In the same operation request to remove patchsets and add patchsets can be specified. Each element of the list consists of the following attribute:			
PatchSet. PatchSetName	The name of a patch set.	1..1024	Character String	N/A
PatchLocation	- Indicates whether to download the patches from the sis server or to pick them up from the OpenScape Voice system at location or to use them from the DVD. This is enum with the options, SIS, OpenScape Voice, or DVD	enum	PatchLocationHIQ PatchLocationDVD PatchLocationSIS	PatchLocationHIQ
PatchSetNodeLocation	Indicates which node has the patchset, in case patchLocation = OpenScape Voice or DVD.	enum	NodeId1 NodeId2	required
sisIP	The IP address of the SIS (Software Installation Server). Required for PatchLocationSIS.	1..1024	Character String	“““
sisUser	The user name with which to log on to the SIS (Software Installation Server). Required for PatchLocationSIS.	1..1024	Character String	“““

Table 842 Upgrade Input Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
sisIP	The password with which to log on to the SIS (Software Installation Server). Required for PatchLocationSIS.	1..1024	Character String	“”

Table 842 Upgrade Input Fields (Sheet 2 of 2)

20.5 Upgrade Second Node

This method initiates an upgrade on the second node of a system. Normally this operation would be executed after the first node has been upgraded successfully. No data parameters are included in the request, since all necessary information has already been sent via the “Upgrade” method.

Table 843 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
11.00.06.08		subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_SPT_TIMEOUT_ERROR subRC_APP_SPT_NO_INIT_ERROR subRC_APP_SPT_SERVER_ERROR

Table 843 Upgrade Second Node Return Codes

Table 844 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to the definition of OperatorId in Section 5.4, “Primitive Data Definitions”, on page 66.			required

Table 844 Upgrade Second Node Input Fields

20.6 Get Upgrade Status

This method returns the current status of an upgrade operation.

Table 845 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
11.00.06.08		subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_SPT_NO_INIT_ERROR subRC_APP_SPT_GEN_ERROR subRC_SYS_SPT_ALLOC_ERROR subRC_APP_SPT_SERVER_ERROR subRC_APP_SPT_NO_INIT_ERROR subRC_APP_SPT_TIMEOUT_ERROR

Table 845 Get Upgrade Status Return Codes

Table 846 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to the definition of OperatorId in Section 5.4, “Primitive Data Definitions”, on page 66.			required

Table 846 Get Upgrade Status Input Fields

Table 847 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
NodeId	Indicates whether the request is for “Node 1” or “Node 2”.	enum	NodeId1 NodeId2	N/A
CEName	The CE node name.	1..1024	Character String	N/A
CEIP	The CE node IP address.	1..1024	Character String	N/A
message	A message string with status or error information.	1..1024	Character String	N/A
percentageCompleted	The percentage of the upgrade process that has been completed (from 0 to 100%).	integer	0..100	N/A
upgradeState	The current state of the upgrade.	enum	UpgradeStateUnknown UpgradeStateNotRunning UpgradeStateInProgress UpgradeStateFailed UpgradeStateFinishedFailed UpgradeStateFinishedSuccessful UpgradeStatePreCommited UpgradeStatePreSecondNodeUpgrade UpgradeStateFallbackFailed UpgradeStateFallbackInProgress UpgradeStateFallbackFinished	N/A
startTime	The date and time when the operation was started.	1..1024	Character String in the format: YYYY-MM-DDTHH:MM:SS	N/A

Table 847 Get Upgrade Status Output Fields (Sheet 1 of 2)

Field Name	Description	Length	Range	Default Value
operatorId	The ID of the operator who initiated the operation.	Please refer to the definition of OperatorId in Section 5.4, “Primitive Data Definitions”, on page 66.		N/A
originatorIP	The IP address of the client that initiated the request.	1..20	Character String in valid IP address format.	N/A
fromPatchSet	The latest patchset information from before the upgrade was started.	1..1024	Character String	N/A
toPatchSet	The target patchset.	1..1024	Character String	N/A
upgradeMethod	Indicates the specified upgrade method.	enum	UpgradeMethodOnline UpgradeMethodRolling UpgradeMethodSMU	N/A

Table 847 Get Upgrade Status Output Fields (Sheet 2 of 2)

20.7 Upgrade Commit

This method finalizes the upgrade by allowing it to be committed. Normally performed after verification of system health, call processing, and other checks are completed on the upgraded system.

Table 848 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
11.00.06.08		subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_SPT_TIMEOUT_ERROR subRC_APP_SPT_NO_INIT_ERROR subRC_APP_SPT_SERVER_ERROR

Table 848 Upgrade Commit Return Codes

Table 849 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to the definition of OperatorId in Section 5.4, “Primitive Data Definitions”, on page 66.			required

Table 849 Upgrade Commit Input Fields

20.8 Upgrade Fallback

This method performs a fallback of the most recent (i.e. the current) upgrade attempt. It is used if errors are reported or manual verification checks fail on the upgraded system. Fallback will (attempt to) uninstall the upgrade on both nodes, if applicable.

Table 850 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
11.00.06.08		subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_APP_SPT_TIMEOUT_ERROR subRC_APP_SPT_NO_INIT_ERROR subRC_APP_SPT_SERVER_ERROR

Table 850 Upgrade Fallback Return Codes

Table 851 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to the definition of OperatorId in Section 5.4, “Primitive Data Definitions”, on page 66.			required

Table 851 Upgrade Fallback Input Fields

20.9 Get Node Health

This method is identical to the Get Node Info request, except that in addition to the information provided, it also includes the results of the RapidStat utility that can be used to report errors or warnings regarding the status of the switch.

Attention: This request may take longer to complete due to the time required to perform the RapidStat inquiry. If a faster response is needed, the GetNodeInfo request should be used.

Table 852 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
11.00.06.08		subRC_SUCCESS subRC_APP_COMM_OPERATOR_INVALID subRC_SYS_SPT_ALLOC_ERROR subRC_APP_SPT_NO_INIT_ERROR subRC_APP_SPT_GEN_ERROR subRC_SYS_SPT_ALLOC_ERROR subRC_APP_SPT_SERVER_ERROR

Table 852 Get Node Health Return Codes

Table 853 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Please refer to the definition of OperatorId in Section 5.4, “Primitive Data Definitions”, on page 66.			required

Table 853 Get Node Health Input Fields

Table 854 lists the *output* fields for this method.

Field Name	Description	Length	Range	Default Value
CEName	The CE (Cluster Element) node name.	1..1024	Character String	N/A
CEIP	The CE node IP address.	1..1024	Character String	N/A
NodeDBRole	Indicates whether the database on the node is currently Primary or Secondary.	enum	RolePrimary RoleSecondary	N/A
NodeId	Indicates whether the node is designated “Node 1” or “Node 2”.	enum	NodeId1 NodeId2	N/A
rtpVersion	The RTP (Resilient Telco Platform) software version currently running on the node.	1..1024	Character String	N/A
patchVersion	The patch set version currently running on the node.	1..1024	Character String	N/A
osType	The operating system type.	1..1024	Character String	N/A
osVersion	The operating system version.	1..1024	Character String	N/A
VIP	Virtual IP address of the cluster.	1..1024	Character String	N/A
dbVersion	The version of the Solid database.	1..1024	Character String	N/A
thirdPartyVersion	The version of third party software on the node.	1..1024	Character String	N/A
RapidStatDataList	A structured list of all warnings and errors reported by the RapidStat tool. Each list element (RapidStatDataEntry) contains the following fields:			
RapidStatDataEntry. RapidStatMessageTy pe	The type of message reported by RapidStat.	enum	RSWarning RSError	N/A
RapidStatDataEntry. RapidStatMessage	The text of the message reported by RapidStat.	1..1024	Character String	N/A

Table 854 Get Node Health Output Fields

20.10 Query Operation Mode

This function will return the own node and partner node Operation Mode state of a cluster system. The following are possible operation mode states:

- Normal
- NormalC

- StandAonePrimary
- StandAloneSync
- StandAloneSecondary
- StandAlone
- ShuttingDown
- Unknown

Table 855 lists the *output* fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		

Table 855 Query Operation Mode Output Fields

20.11 Modify Operation Mode

Table 856 lists the *input* fields for this method.

Field Name	Description	Length	Range/Values	Default Value
OperatorId	The operator issuing the command.	Please refer to the definition of Operator ID in Section 5.4.2, “Field Definitions”, on page 67.		
NewOpMode	Mandatory field. Specifies the operation mode status of the node of the cluster to be modified. The craftsperson can manually change the operation mode of a node in a cluster only if the two nodes cannot communicate with one another over the cross-channel. The allowed from states are: • StandAlonePrimary • StandAloneSecondary The manual modification is allows for only the own node because there is no communication with the partner node. Required field	enum	OpModeNormal OpModeStandAlonePrimary OpModeStandAloneSecondary	

Table 856 Modify Operation Mode Input Fields

20.12 Get Database Role

This Web Service will retrieve the Database Role of the H8K. The new API will be used by Assistant to determine which H8K node hosts the Primary Database connection so as to direct Mass Provisioning Commands/Operations in that node only.

Table 857 lists the return codes for this method.

WSDL Version	Obsolete in WSDL Version	Return Codes
V5.00.01.02		subRC_APP_COMM_OPERATOR_INVALID subRC_SYS_SPT_ALLOC_ERROR subRC_APP_DSA_DBROLE_RETRIEVAL_FAIL

Table 857 Get Database Role Return Codes

Table 858 lists the *input* fields for this method.

Field Name	Description	Length	Range	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)

Table 858 Get Database Role Input Fields

Table 859 lists the *output* fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.
GetDBRoleResult	The Database Node Result returns an enumerator indicating the possible values of the Database Connection: The possible values are: **RolePrimary, RoleSecondary**

Table 859 Get Database Role Output Fields

21 Traffic Types and Classes of Restrictions

FRN3454 introduced a new concept for the existing functionality of Traffic Types, where additional Traffic Types shall be able to be provisioned on System Level (Global Numbering Plan) and be used by Destination Codes and Code Indices for translation.

A collection of Traffic Types shall be able to be grouped together under a new Provisioning Entity called Class of Restriction. Classes of restrictions shall be used by the features TRS (Subscribers, FeatureProfile, Endpoint Profile), CFR (Subscribers, Features Profile), AuthCode (Subscribers, Features Profile), in order to determine whether to disallow the call or request an authorization code before allowing the call to go through.

21.1 Create Traffic Type

Soap Server shall be enhanced in order to support the provisioning of new Traffic Types in the Default Number Plan (Global Translation and Routing Tab of Assistant). There will exist two Variants of Traffic Types: Emergency and Generic. The administrator shall be able to provision 127 additional Generic Traffic Types but shall not be able to provision additional Emergency Traffic Types. The Variant will not be exposed in this interface.

Validations to be performed by Soap Server are: Presence of Mandatory Traffic Type Name, Usage of Invalid characters, Length of Traffic Type Name, Already Existing Traffic Type Name.

Effective starting with V6, the administrator can enter a TrafficType value/id, which shall be associated with the TrafficType Name. The new field is optional. Validations shall be performed by SDAL/SOAP and appropriate error messages shall be returned whenever:

- Invalid Trafifc Type ID/Value has been selected
- The Traffic Type ID/Value is already being used

Table 860 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.03.01	subRC_APP_COMM_OPERATOR_INVALID subRC_APP_TRAFFIC_TYPE_NAME_LEN_INVALID subRC_APP_TTNAME_SDAL_INITIALIZATION_ERROR subRC_APP_TTNAME_ALREADY_EXISTS subRC_APP_TT_TYPE_LIMIT_EXCEED subRC_APP_GENERAL_SDAL_ERROR

Table 860 Create Traffic Type Return Codes

Table 861 lists the *input* fields for this method.

Field Name	Required	Description	Type	Max Length (Range)	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
TrafficTypeName	Yes	I/F Traffic Type Name	Char array	32 (1…32)	N/A
Traffic Type Id	No	I/F Traffic Type ID	int		N/A

Table 861 Create Traffic Type Input Fields

Table 862 lists the *output* fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed above.

Table 862 Create Traffic Type Output Fields

21.2 Modify Traffic Type

Soap Server shall be enhanced in order to support the modification of the Name of an already provisioned Traffic Type.

Validations to be performed by Soap Server are in addition to the ones in the Create Interface: Already existing New Traffic Type Name, Non existing Traffic Type for the Old Traffic Type Name.

Table 863 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.03.01	subRC_APP_COMM_OPERATOR_INVALID subRC_APP_TRAFFIC_TYPE_NAME_LEN_INVALID subRC_APP_NEW_TRAFFIC_TYPE_NAME_NOT_PROVIDED subRC_APP_TTNAME_SDAL_INITIALIZATION_ERROR subRC_APP_TTNAME_NON_EXISTING subRC_APP_TTNAME_ALREADY_EXISTS subRC_APP_GENERAL_SDAL_ERROR

Table 863 Modify Traffic Type Return Codes

Table 864 lists the *input* fields for this method.

Field Name	Required	Description	Type	Max Length (Range)	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
TrafficTypeName	Yes	I/F Traffic Type Name	Char array	32 (1…32)	N/A
NewTrafficTypeName	Yes	I/F New Traffic Type Name	Char array	32 (1…32)	N/A

Table 864 Modify Traffic Type Input Fields

Table 865 lists the *output* fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed above.

Table 865 Modify Traffic Type Output Fields

21.3 Delete Traffic Type

Soap Server shall be enhanced to support the deletion of a Generic Traffic Type. Validations to be performed are: Non Existing Traffic Type, Specific Traffic Type is being used in translation by Destination Codes/Code Indices; Specific Traffic Type is being used by a Class of Restriction or Destination Codes.

Table 866 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.03.01	subRC_APP_COMM_OPERATOR_INVALID subRC_APP_TRAFFIC_TYPE_NAME_LEN_INVALID subRC_APP_TTNAME_SDAL_INITIALIZATION_ERROR subRC_APP_TTNAME_REFERENCED_BY_XDM subRC_APP_TTNAME_REFERENCED_BY_COR subRC_APP_TTNAME_EMERGENCY subRC_APP_TTNAME_NON_EXISTING subRC_APP_GENERAL_SDAL_ERROR

Table 866 Delete Traffic Type Return Codes

Table 867 lists the *input* fields for this method.

Field Name	Required	Description	Type	Max Length (Range)	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
TrafficTypeName	Yes	I/F Traffic Type Name	Char array	32 (1…32)	N/A

Table 867 Delete Traffic Type Input Fields

Table 868 lists the *output* fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed above.

Table 868 Delete Traffic Type Output Fields

21.4 Query Traffic Type

Soap Server supports the List retrieval of Traffic Type Names together with an indication of whether it is a generic or an emergency Traffic type. Filtering on TrafficType Name, Ascending or Descending Order as well as Paging Information shall be supported.

Effective with V6 the new TrafficType ID/value field is returned for each TrafficType Name. Sorting of the retrieved list based on TrafficType ID/Value shall also be supported along with the TrafficType Name.

lists the return codes for this method.

WSDL Version	Return Codes
V5.00.03.01	subRC_APP_COMM_OPERATOR_INVALID subRC_APP_TRAFFIC_TYPE_NAME_LEN_INVALID subRC_APP_TTNAME_SDAL_INITIALIZATION_ERROR subRC_APP_TT_QUERY_LIST_FAIL

Table 869 Query Traffic Type Return Codes

Table 870 lists the *input* fields for this method.

Field Name	Required	Description	Type	Max Length (Range)	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
TrafficTypeName	No	I/F Traffic Type Name	Char array	32 (1…32)	N/A
TrafficTypevariant	No	I/F Traffic Type	Enumerator	-	-
Page Number	No	number of the retrieved result set	Integer	-	-
Page Size	No	Page size of the retrieved result set	Integer	[1-128]	-
Sorting	No	The sort order of the retrieved result set	Enumerator	Ascending Descending	Ascending
TTSortType	No	Sort filter criteria for the Traffic types	Enumerator	TTSortByName, TTSortByValue	TTSortByNam e

Table 870 Query Traffic Type Input Fields

Table 871 lists the *output* fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed above.
PaginatedInfo	A common paging structure containing Current Page/ Num of Pages / Page Size etc.
QueryTrafficType List	A list containing Traffic Type Names with an indication of Generic or Emergency

Table 871 Query Traffic Type Output Fields

21.5 Create Class Of Restrictions

Soap Server shall be enhanced in order to support the provisioning of a Class of Restriction in the Default Number Plan (Global Translation and Routing Tab of Assistant). The Class of Restriction shall consist of a List with a combination of Generic Traffic Types.

Soap Server shall provide the name of the Class of restriction along with the names of Traffic Types found in the list in SDAL, which shall set the values in the bitmaps.

Validations to be performed: Presence of mandatory COR Name, COR Name Length validation, Already existing COR Name, Traffic Type Name does not exist; List contains Traffic Type of Variant Emergency.

Table 872 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.03.01	subRC_APP_COMM_OPERATOR_INVALID subRC_APP_TRAFFIC_TYPE_NAME_LEN_INVALID subRC_APP_TTNAME_SDAL_INITIALIZATION_ERROR subRC_APP_TT_QUERY_LIST_FAIL

Table 872 Create COR Return Codes

Table 873 lists the *input* fields for this method.

Field Name	Required	Description	Type	Max Length (Range)	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
CORName	Yes	I/F COR Name	Char array	32 (1…32)	N/A
TrafficTypeNamesList	No	The list of Traffic Types to be associated with COR			

Table 873 Create COR Input Fields

Table 874 lists the *output* fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed above.

Table 874 Create COR Output Fields

21.6 Modify Class Of Restrictions

Soap Server shall be enhanced in order to support the modification of the name of a Class of Restriction if a new name is provided. Moreover, modification of the Traffic Type List associated with the COR shall be supported.

It is important to note that Assistant shall have to always send the List of Traffic Types when using the ModifyClassofRestriction API. Based on the List of Traffic Types in the request, the old bitmap shall be reset and set again.

Validations to be performed are in addition to the ones in the Create Interface: Already existing New COR Name, Not existing Traffic Types for COR Name, Non Existing Traffic Types, List of Traffic Types contains an Emergency one.

Table 875 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.03.01	subRC_APP_COMM_OPERATOR_INVALID subRC_APP_TRAFFIC_TYPE_NAME_LEN_INVALID subRC_APP_TTNAME_SDAL_INITIALIZATION_ERROR subRC_APP_TT_QUERY_LIST_FAIL

Table 875 Modify COR Return Codes

Table 876 lists the *input* fields for this method.

Field Name	Required	Description	Type	Max Length (Range)	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
CORName	Yes	I/F COR Name	Char array	32 (1…32)	N/A
NewCORName	Yes	I/F New COR Type Name	Char array	32 (1…32)	N/A
TrafficTypeNamesList	No	The list of Traffic Types to be associated with COR			

Table 876 Modify COR Input Fields

Table 877 lists the *output* fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed above.Return code list is displayed above.

Table 877 Modify COR Output Fields

21.7 Delete Class Of Restrictions

Soap Server shall be enhanced in order to support the deletion of a Class of restriction. Validations to be performed are: Non Existing Class of Restriction, Class of Restriction is in use by services.

Table 878 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.03.01	subRC_APP_COMM_OPERATOR_INVALID subRC_APP_COR_NAME_LEN_INVALID subRC_APP_CORNAME_SDAL_INITIALIZATION_ERROR subRC_APP_CORNAME_NON_EXISTING subRC_APP_CORNAME_REFERENCED_BY_SERVICES subRC_APP_GENERAL_SDAL_ERROR

Table 878 COR Input Fields

Table 879 lists the *input* fields for this method.

Field Name	Required	Description	Type	Max Length (Range)	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
CORName	Yes	I/F COR Name	Char array	32 (1…32)	N/A

Table 879 Delete COR Input Fields

Table 880 lists the *output* fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed above.

Table 880 Delete COR Output Fields

21.8 Query Class Of Restrictions

Soap Server shall be enhanced in order to support the retrieval of the list of Class of Restriction Names. The interface shall support paging as well as paging/sorting based on the COR Name

Table 881 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.03.01	subRC_APP_COMM_OPERATOR_INVALID subRC_APP_COR_NAME_LEN_INVALID subRC_APP_CORNAME_SDAL_INITIALIZATION_ERROR subRC_APP_COR_NAME_QUERY_LIST_FAIL

Table 881 Query COR Names List Input Fields

Table 882 lists the *input* fields for this method.

Field Name	Required	Description	Type	Max Length (Range)	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
CORName	No	I/F COR Name	Char array	32 (1…32)	N/A
Page Number	No	number of the retrieved result set	Integer	-	-
Page Size	No	Page size of the retrieved result set	Integer	-	-
Sorting	No	The sort order of the retrieved result set	Enumerator	Ascending Descending	Ascending

Table 882 Query COR Names List Input Fields

Table 883 lists the *output* fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed above.
PaginatedInfo	A common paging structure containing Current Page/ Num of Pages / Page Size etc.
QueryCORNames List	A list containing COR Names

Table 883 Query COR Names List Output Fields

21.9 Query Class Of Restrictions Info List

Soap Server shall be enhanced in order to support the query for the values of the Traffic Types in a specific Class of Restriction. Paging shall not be supported since a list of up to 128 Traffic Types should be fine. Sorting based on Traffic Types shall be supported but filtering shall not be applicable.

Table 884 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.03.01	subRC_APP_COMM_OPERATOR_INVALID subRC_APP_COR_NAME_LEN_INVALID subRC_APP_CORNAME_SDAL_INITIALIZATION_ERROR subRC_APP_COR_NAME_QUERY_LIST_FAIL

Table 884 Query COR Info List Input Fields

Table 885 lists the *input* fields for this method.

Field Name	Required	Description	Type	Max Length (Range)	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A

Table 885 Query COR Info List Input Fields (Sheet 1 of 2)

Field Name	Required	Description	Type	Max Length (Range)	Default Value
CORName	Yes	I/F COR Name	Char array	32 (1…32)	N/A
Sorting	No	The sort order of the retrieved result set	Enumerator	Ascending Descending	Ascending

Table 885 Query COR Info List Input Fields (Sheet 2 of 2)

Table 886 lists the *output* fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed above.
CORName	The name of the COR for which TrafficType List shall be retrieved
TrafficTypeNamesList	The list of Traffic Types to be associated with COR

Table 886 Query COR Info List Output Fields

21.10 Query COR Usage List

Soap Server shall be enhanced in order to support the retrieval of the list of Subscribers and/or Endpoint Profiles that are using the Class of Restriction through the Call Forward Restrictions (CFR), the AuthCode and TRS services. In the case of TRS, a standard as well an alternate Class of Restriction can be provisioned. The Query Interface shall display the usage of both Standard and Alternate Class of Restrictions.

Table 887 lists the return codes for this method.

WSDL Version	Return Codes
V5.00.03.01	subRC_APP_COMM_OPERATOR_INVALID subRC_APP_COR_NAME_LEN_INVALID subRC_APP_CORNAME_SDAL_INITIALIZATION_ERROR subRC_APP_CORNAME_NON_EXISTING

Table 887 Query COR Usage List Input Fields

Table 888 lists the *input* fields for this method.

Field Name	Required	Description	Type	Max Length (Range)	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
CORName	Yes	I/F COR Name	Char array	32 (1…32)	N/A
Sorting	No	The sort order of the retrieved result set	Enumerator	Ascending Descending	Ascending
Page Number	No	number of the retrieved result set	Integer	-	-

Table 888 Query COR Usage List Input Fields (Sheet 1 of 2)

Field Name	Required	Description	Type	Max Length (Range)	Default Value
Page Size	No	Page size of the retrieved result set	Integer	-	-

Table 888 Query COR Usage List Input Fields (Sheet 2 of 2)

Table 889 lists the *output* fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed above.
PaginatedInfo	A common paging structure containing Current Page/ Num of Pages / Page Size etc.
CORUsageList	The list of Usage Information for the requested COR. The UsageListEntry shall return: UsageName, UsageType (UndefinedUsageType, Subscriber, EndpointProfile),and Boolean indication for: StandardTollRestrictions, AlternateTollRestrictions, CallForwardRestrictions, AuthorizationCodes

Table 889 Query COR Usage List Output Fields

21.11 Query COR Group Key List

The Authorization Code Toll Restriction Override service is a system wide service that allows administrators to create groups of authorization codes. For each group a class of restriction can be defined that contains the toll restricted traffic types that are overridden. Authorization Code Groups are formed by simply using the leading digits of the authorization codes as the key that identifies a group.

This feature is activated system-wide by setting the `Srx/Main/AuthTRSOverride` RTP parameter to true. The group keys and Class of restriction associations are stored in the `Srx/Main/AuthTRSOverrideValues` RTP parameter in the following format:

```
<GroupKey1>:<COR1>;<GroupKey2>:<COR2>; ….
```

Once activated, all off-net destinations marked with a generic traffic type (i.e. non-emergency traffic type) will be restricted unless the traffic type is listed in a class of restriction assigned to the authorization code group to which the authorization code entered on the call belongs.

After the validation of an entered authorization code, the Authorization Code Toll Restriction Override service when activated, matches the entered authorization code against the Authorization Code Group keys.

- If no key matches, then no toll restrictions will be overridden, i.e. the currently restricted traffic types will remain in effect and the call will be released as if the call were restricted by the toll restriction service.

- If 1 key matches, then the traffic type of the call will be looked up in the class of restriction that is associated to the key. If it is listed, then the call may proceed. If it is not listed, the call will be released as if the call were restricted by the toll restriction service.
- If more than 1 key matches, the best matching key (on key length) is used. The traffic type of the call will be looked up in the class of restriction that is associated to the matching key. If it is listed, then the call may proceed. If it is not listed, the call will be released as if the call were restricted by the toll restriction service.

The QueryCORGroupKeyList API returns a list of the Authorization Code Group Keys that are associated with the requested Class of Restriction.

Table 890 lists the *input* fields for this method.

Field Name	Required	Description	Type	Max Length (Range)	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
CORName	Yes	I/F COR Name	Char array	32 (1…32)	N/A
Sorting	No	The sort order of the retrieved result set	Enumerator	Ascending Descending	Ascending
Page Number	No	number of the retrieved result set	Integer	-	-
Page Size	No	Page size of the retrieved result set	Integer	-	-

Table 890 Query COR Group Key List Input Fields

Table 891 lists the *output* fields for this method.

Field Name	Description
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed above.
PaginatedInfo	A common paging structure containing Current Page/ Num of Pages / Page Size etc.
CORGroupKeyList	The list of COR Group Key information for the requested COR Group.

Table 891 Query COR Group Key List Output Fields

21.12 CSTA Applications

SOAP supports the creation, modification, query and deletion of CSTA applications, the retrieval of the list of provisioned applications as well as the list of active devices for each application. When a CSTA application is created,

modified or removed from the system, SOAP the associated packet filter rules are updated. Up to 256 applications are allowed to be provisioned. CSTA application can be provisioned using IPv4 addresses only.

The following APIs are used for the provisioning of CSTA applications:

- GetCSTAApplicationList
- CreateCSTAApplication
- ModifyCSTAApplication
- QueryCSTAApplication
- GetCSTAActiveDeviceList
- DeleteCSTAApplication
- GetCSTASMInfo

21.12.1 GetCSTAApplicationList

The *GetCSTAApplicationList* API (starting in V7.00.06.01) allows the retrieval of the list of CSTA applications provisioned in the system.

Table 892 lists the *input* fields of *GetCSTAApplicationListRequest*:

Field	Description	Required	Type	Default Value [Range]
OperatorId	I/F operator	yes	Char array	NA [1…40]
ApplicationType	Allows filtering of the result per application type or quering for all application	no	Enumeration	CSTAAppTypeAll CSTAAppTypeOpenscape CSTAAppTypeThirdParty CSTAAppTypeTelefonica CSTAAppTypeUnknown
Paging	Standard paging structure			
SortType	The sort criterion applied to the request.	no	Enumeration	CSTAAppSortTypeApplicationType CSTAAppSortTypeIP CSTAAppSortTypeNameCSTAAppSortTypeConnected
SortOrder	Standard sorting structure			

Table 892 GetCSTAApplicationListRequest fields

Table 893 lists the *input* fields of *GetCSTAApplicationListResult*:

Field	Description	Type	Range - Length
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.		
PaginatedInfo	A common paging structure containing Current Page/ Num of Pages / Page Size etc.		
ApplicationList	A list of ApplicationData elements		
(ApplicationList) ApplicationData	A structure containing the following data		
(ApplicationData) ApplicationName	Provides a description of the CSTA application. ApplicationName is a unique identifier. If a CSTA application is connected before it is provisioned it has a default name which is the application's IP address.	Char array	1..64
(ApplicationData) IPAddress	The (unique) IP of the application in IPv4 format.	Char array	1..45
(ApplicationData) ApplicationType	The type of the application. Value Unkown is set to an application if the application connects before provisioned.	Enumeration	CSTAAppTypeOpenscape CSTAAppTypeThidParty CSTAAppTypeTelefonica CSTAAppTypeUnknown
(ApplicationData) Connected	Indicates if the application is currently connected. This means that the application has established a CSTA over TCP or CSTA over HTTP/SOAP connection	Boolean	False, true
(ApplicationData) Monitors	The number of active CSTA monitored devices	Unsigned Integer	-
(ApplicationData) RouteRegistrations	The number of active CSTA route registration devices	Unsigned Integer	-
Totals	A structure containing the following data		
(Totals) RouteRegistrations	The total number of active CSTA monitored devices	Unsigned Integer	-
(Totals) Monitors	The total number of active CSTA monitored devices	Unsigned Integer	-

Table 893 GetCSTAApplicationListResult fields

The following table shows the possible error codes of *GetCSTAApplicationList* API:

WSDL Version	Return Codes	Message
V7.00.06.01	subRC_APP_COMM_OPERATOR_INVALID	Invalid Operator name. Length invalid or characters not allowed.
	subRC_DB_GENERAL_SDAL_DB_ERROR	SDAL DB General Error
	subRC_APP_GENERAL_SDAL_ERROR	General SDAL Error

Table 894 GetCSTAApplicationList Return Codes

21.12.2 CreateCSTAApplication

The *CreateCSTAApplication* API (starting with V7.00.06.01) allows the provisioning of new CSTA applications. The creation of an application with existing IP address or name is blocked. Up to 256 applications are allowed to be created.

Table 895 lists the *input* fields of *CreateCSTAApplicationRequest*:

Field	Description	Required	Type	Default Value [Range]
OperatorId	I/F operator	yes	Char array	NA [1…40]
Identification	A required structure containing the following data			
(Identification) ApplicationIPAddress	The (unique) IP of the application in IPv4 format. Maximum length is 45 characters to allow future use of IPv6 addressing.	yes	Char array	1..45
(Identification) ApplicationName	Provides a description of the CSTA application. The ApplicationName is a unique identifier of an application. If a CSTA application is connected before they are provisioned it has a default name which is the application's IP address. If the application type is Telefonica then this name is copied to the Application ID located in the Operational Data section.	yes	Char array	1..64
(Identification) ApplicationType	CSTAAppTypeOpenscape CSTAAppTypeThidParty CSTAAppTypeTelefonica	no	Enumeration	CSTAAppTypeOpenscape (default) CSTAAppTypeThidParty CSTAAppTypeTelefonica
SOAPHeartbeat	A structure containing the following data.			

Table 895 CreateCSTAApplicationRequest fields

(SOAPHeartbeat) SOAPListenerPort	Port that applications use when connecting to OpenScape Voice via CSTA over HTTP/SOAP. CSTA over HTTP/SOAP uses same IP address as CSTA over TCP.	no	Unsigned Integer	Default: 7780 80, 1024 - 65535
(SOAPHeartbeat) HeartbeatTimer	System Status Heartbeat timer interval value. A value of 0 disables the heartbeat function.	no	Unsigned Integer	Default: 0 0-3600
(SOAPHeartbeat) HeartbeatWatchdog	System Status Heartbeat watchdog timer value.	no	Unsigned Integer	Default: 0 70..360
(SOAPHeartbeat) HeartbeatRetryCount	System Status Heartbeat retry value.	no	Unsigned Integer	Default: 0 0..5
OperationalData	A structure containing the following data. Applicable only to application of type ApplicationTypeTelefonica.			
(OperationalData) ApplicationID	The Application ID. If this field is not provided then it corresponds to the ApplicationName of the Identity section.	no	Char array	1..64
(OperationalData) EnterpriseID	Telefonica Enterprise Identification prefixed to subscriber deviceIDs presented to Telefonica application.	no	Char array	Default: 000000 6..8
(OperationalData) PASCODE	Telefonica prefixes this code to the routedSelected parameter when routing to the mobile network VPN. CRS uses this parameter to determine when to invoke Route End with RouteEndReason = endCstaDistribution.	no	Char array	Default: 000000 1..8
(OperationalData) OutsideAccess	Telefonica outside access code for external number conversion.	no	Char array	1..30
(OperationalData) BGName	Used to convert subscriber extension to fully qualified number via new mapping function.	no	Char array	1..30
(OperationalData) NPName	Used to convert subscriber extension to fully qualified number via new mapping function.	no	Char array	1..30

Table 895 CreateCSTAApplicationRequest fields

Table 896 lists the fields of *CreateCSTAApplicationResult*:

Field	Description	Type	Range - Length
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.		

Table 896 CreateCSTAApplicationResult fields

The following table shows the possible error codes of *CreateCSTAApplication* API:

WSDL Version	Return Codes	Message
V7.00.06.01	subRC_APP_COMM_OPERATOR_INVALID	Invalid Operator name. Length invalid or characters not allowed.
	subRC_APP_CSTA_INVALID_VALUE	Invalid Value provided
	subRC_APP_CSTA_INVALID_APP_IP	Invalid Value provided
	subRC_APP_CSTA_INVALID_APP_TYPE	Invalid value provided: Application Type. Allowed values: Openscape, ThirdParty, Telefonica
	subRC_APP_CSTA_INVALID_APP_NAME	Invalid Value provided
	subRC_APP_CSTA_INVALID_OPERATION_DATA	Operational data may only be provided for Applications of type Telefonica Operational data are required for Applications of type Telefonica
	subRC_APP_CSTA_BG_NP_REQUIRED	At least one of BGName,NPName must be provided for Applications of type Telefonica
	subRC_APP_CSTA_APP_IP_EXISTS	A CSTA Application with the specified IP Address already exists
	subRC_APP_CSTA_APP_NAME_EXISTS	A CSTA Application with the specified Name already exists
	subRC_APP_CSTA_APP_MAX_APPS_REACHED	A CSTA Application with the specified Name already exists
	subRC_DB_GENERAL_SDAL_DB_ERROR	A CSTA Application with the specified Name already exists
	subRC_APP_GENERAL_SDAL_ERROR	A CSTA Application with the specified Name already exists
	subRC_APP_CSTA_BG_NP_MUTUALLY_EXCLUSIVE	BGName and NPName are mutually exclusive for Applications of type Telefonica
	subRC_APP_CSTA_PFR_UPDATE_ERROR	secPacketFilter.AddCstaAppRule Failed
	subRC_APP_CSTA_APP_SOASF_RTP_SEND_ERROR	Failed to send message to SOASF

Table 897 CreateCSTAApplication Return Codes

21.12.3 ModifyCSTAApplication

The *ModifyCSTAApplication* API (starting with V7.00.06.01) allows the modification of a CSTA application. If the IP address of an application is changed SOAP will update the packets filtering rules. SOAP does not allow the modification of the ApplicationType of an existing application, except the transition from type *Unknown* to other types.

Table 898 lists the fields of *ModifyCSTAApplicationRequest*:

Field	Description	Required	Type	Default Value [Range]
OperatorId	I/F operator	yes	Char array	NA [1…40]
ApplicationIPAddress	The (unique) IP of the application to be updated in IPv4 format. Maximum length is 45 characters to allow future use of IPv6 addressing.	yes	Char array	1..45
Identification	A required structure containing the following data			
(Identification) ApplicationIPAddress	The new IP of the application in IPv4 format. This IP must unique among applications.	yes	Char array	1..45
(Identification) ApplicationName	Provides a description of the CSTA application. ApplicationName is a unique identifier. If a CSTA application is connected before they are provisioned it has a default name which is the application's IP address. If the application type is Telefonica then this name is copied to the Application ID located in the Operational Data section.	yes	Char array	1..64
(Identification) ApplicationType	The type of the application. Application type may be: CSTAAppTypeOpenscape (default) CSTAAppTypeThidParty CSTAAppTypeTelefonica	no	Enumeratio n	CSTAAppTypeOpenscap e (default) CSTAAppTypeThidParty CSTAAppTypeTelefonica
SOAPHeartbeat	A structure containing the following data.			
(SOAPHeartbeat) SOAPListenerPort	Port that applications use when connecting to OpenScape Voice via CSTA over HTTP/SOAP. CSTA over HTTP/SOAP uses same IP address as CSTA over TCP.	no	Unsigned Integer	Default: 7780 80, 1024 - 65535
(SOAPHeartbeat) HeartbeatTimer	System Status Heartbeat timer interval value. A value of 0 disables the heartbeat function.	no	Unsigned Integer	Default: 0 0-3600
(SOAPHeartbeat) HeartbeatWatchdog	System Status Heartbeat watchdog timer value.	no	Unsigned Integer	Default: 0 70..360
(SOAPHeartbeat) HeartbeatRetryCount	System Status Heartbeat retry value.	no	Unsigned Integer	Default: 0 0..5
OperationalData	A structure containing the following data. Applicable only to application of type ApplicationTypeTelefonica.			
(OperationalData) ApplicationID	The Application ID. If this field is not provided then it corresponds to the ApplicationName of the Identity section.	no	Char array	1..64
(OperationalData) EnterpriseID	Telefonica Enterprise Identification prefixed to subscriber deviceIDs presented to Telefonica application.	no	Char array	Default: 000000 6..8

Table 898 ModifyCSTAApplicationRequest fields

(OperationalData) PASCODE	Telefonica prefixes this code to the routedSelected parameter when routing to the mobile network VPN. CRS uses this parameter to determine when to invoke Route End with RouteEndReason = endCstaDistribution.	no	Char array	Default: 000000 1..8
(OperationalData) OutsideAccess	Telefonica outside access code for external number conversion.	no	Char array	1..30
(OperationalData) BGName	Used to convert subscriber extension to fully qualified number via new mapping function.	no	Char array	1..30
(OperationalData) NPName	Used to convert subscriber extension to fully qualified number via new mapping function.	no	Char array	1..30

Table 898 ModifyCSTAApplicationRequest fields

Table 899 lists the fields of *ModifyCSTAApplicationResult*:

Field	Description	Type	Range - Length
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.		

Table 899 ModifyCSTAApplicationResult fields

The following table shows the possible error codes of *ModifyCSTAApplication* API:

WSDL Version	Return Codes	Message
V7.00.06.01	subRC_APP_COMM_OPERATOR_INVALID	Invalid Operator name. Length invalid or characters not allowed.
	subRC_APP_CSTA_INVALID_VALUE	Invalid Value provided
	subRC_APP_CSTA_INVALID_APP_IP	Invalid Value provided
	subRC_APP_CSTA_INVALID_APP_TYPE	Invalid value provided: Application Type. Allowed values: Openscape, ThirdParty, Telefonica
	subRC_APP_CSTA_INVALID_APP_NAME	Invalid Value provided
	subRC_APP_CSTA_INVALID_OPERATION_DATA	Operational data may only be provided for Applications of type Telefonica Operational data are required for Applications of type Telefonica
	subRC_APP_CSTA_BG_NP_REQUIRED	At least one of BGName,NPName must be provided for Applications of type Telefonica
	subRC_APP_CSTA_APP_IP_EXISTS	A CSTA Application with the specified IP Address already exists
	subRC_APP_CSTA_APP_NAME_EXISTS	A CSTA Application with the specified Name already exists
	subRC_DB_GENERAL_SDAL_DB_ERROR	A CSTA Application with the specified Name already exists
	subRC_APP_GENERAL_SDAL_ERROR	A CSTA Application with the specified Name already exists
	subRC_APP_CSTA_BG_NP_MUTUALLY_EXCLUSIVE	BGName and NPName are mutually exclusive for Applications of type Telefonica
	subRC_APP_CSTA_PFR_UPDATE_ERROR	secPacketFilter.ModifyCstaAppRule Failed
	subRC_APP_CSTA_APP_SOASF_RTP_SEND_ERROR	Failed to send message to SOASF
	subRC_APP_CSTA_APP_NOT_PRESENT	No CSTA Application with the specified IP Address exists
	subRC_APP_CSTA_APP_MOD_FAILED_MONITORED_DEVICES_EXIST	Failed to Modify CSTA Application. Active Monitored devices exist
	subRC_APP_CSTA_APP_MOD_FAILED_ROUTE_REG_DEVICES_EXIST	Failed to Modify CSTA Application. Route Registered devices exist
	subRC_APP_CSTA_APP_MOD_FAILED_CONNECTED	Failed to Modify CSTA Application. CSTA Application is connected

Table 900 ModifyCSTAApplication Return Codes

21.12.4 QueryCSTAApplication

The *QueryCSTAApplication* API (starting with V7.00.06.01) allows the retrieval of the data of a CSTA application provisioned in the system.

Table 901 lists the fields of *QueryCSTAApplicationRequest*:

Field	Description	Required	Type	Default Value [Range]
OperatorId	I/F operator	yes	Char array	NA [1…40]
IPAddress	The (unique) IP of the application to be queried in IPv4 format.	yes	Char array	1..45

Table 901 QueryCSTAApplicationRequest Fields

Table 902 lists the fields of *QueryCSTAApplicationResult*:

Field	Description	Type	Range - Length
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.		
Identification	A structure containing the following data		
(Identification) ApplicationIP	The IP of the application IPv4 format.	Char array	1..45
(Identification) ApplicationName	The name of the CSTA application.	Char array	1..64
(Identification) ApplicationType	The type of the application. Application type may be: ApplicationTypeOpenscape ApplicationTypeThidParty ApplicationTypeTelefonica ApplicationTypeUnkown	Enumeration	ApplicationTypeOpenscape ApplicationTypeThidParty ApplicationTypeTelefonica ApplicationTypeUnkown
SOAPHeartbeat	A structure containing the following data.		
(SOAPHeartbeat) SOAPListenerPort	Port that applications use when connecting to OpenScape Voice via CSTA over HTTP/SOAP. CSTA over HTTP/SOAP uses same IP address as CSTA over TCP.	Unsigned Integer	80, 1024 - 65535
(SOAPHeartbeat) HeartbeatTimer	System Status Heartbeat timer interval value. A value of 0 disables the heartbeat function.	Unsigned Integer	0-3600
(SOAPHeartbeat) HeartbeatWatchdog	System Status Heartbeat watchdog timer value.	Unsigned Integer	70..360
(SOAPHeartbeat) HeartbeatRetryCount	System Status Heartbeat retry value.	Unsigned Integer	0..5
OperationalData	A structure containing the following data. Applicable only to application of type ApplicationTypeTelefonica.		

Table 902 QueryCSTAApplicationResult Fields

(OperationalData) ApplicationID	The Application ID. If this field is not provided then it corresponds to the ApplicationName of the Identity section.	Char array	1..64
(OperationalData) EnterpriseID	Telefonica Enterprise Identification prefixed to subscriber deviceIDs presented to Telefonica application.	Char array	6..8
(OperationalData) PASCODE	Telefonica prefixes this code to the routedSelected parameter when routing to the mobile network VPN. CRS uses this parameter to determine when to invoke Route End with RouteEndReason = endCstaDistribution.	Char array	1..8
(OperationalData) OutsideAccess	Telefonica outside access code for external number conversion.	Char array	1..30
(OperationalData) BGName	Used to convert subscriber extension to fully qualified number via new mapping function.	Char array	1..30
(OperationalData) NPName	Used to convert subscriber extension to fully qualified number via new mapping function.	Char array	1..30
ApplicationStatus	A structure containing the following data		
(ApplicationStatus) Status	The SOAP heartbeat status.	Enumeration	Active Inactive
(ApplicationStatus) StatusLastUpdate	Date and time of last update of SOAP heartbeat status.	Char array	0..256
(ApplicationStatus) ConnectionStatus	The application connection status.	Enumeration	CSTAAppConnectionStatusDown CSTAAppConnectionStatusUp CSTAAppConnectionStatusFailed CSTAAppConnectionStatusRetrieved
(ApplicationStatus) ConnectionStatus LastUpdate	Date and time of last update of application connection status.	Char array	0..256
(ApplicationStatus) SessionStatus	The status of the application session service heartbeat.	Enumeration	Active Inactive
(ApplicationStatus) SessionStatus LastUpdate	Date and time of last update of the status of the application session service heartbeat.	Char array	0..256

Table 902 QueryCSTAApplicationResult Fields

SupportedWebService Names	A list of WebServiceName elements		
(SupportedWebService Names) WebServiceName	Supported web service name	Char array	1..64

Table 902 QueryCSTAApplicationResult Fields

The following table shows the possible error codes of *QueryCSTAApplication* API:

WSDL Version	Return Codes	Message
V7.00.06.01	subRC_APP_COMM_OPERATOR_INVALID	Invalid Operator name. Length invalid or characters not allowed.
	subRC_APP_CSTA_APP_NOT_PRESENT	No CSTA Application with the specified IP Address exists
	subRC_APP_GENERAL_SDAL_ERROR	General SDAL Error

Table 903 QueryCSTAApplication Return Codes

21.12.5 GetCSTAActiveDeviceList

The GetCSTAActiveDeviceList API (starting with V7.00.06.01) allows the retrieval of the list of subscriber DNs with active CSTA monitors or route registrations for a CSTA application.

Table 904 lists the fields of *GetCSTAActiveDeviceListRequest*:

Field	Description	Required	Type	Default Value [Range]
OperatorId	I/F operator	yes	Char array	NA [1…40]
ApplicationIP	The (unique) IP of the application to be queried for active devices in IPv4 format.	yes	Char array	1..45
DeviceID	Allows filtering for a Service ID. Supports '*' and '?' wildcards.	no	Char array	0..30
Paging	Standard paging structure			

Table 904 GetCSTAActiveDeviceListRequest Fields

DeviceFilter	Allowes filtering for devices with active CSTA monitors, active CSTA route registrations, subscriber devices only, NID devices only or all devices.	no	Enumeration	CSTADeviceFilterAll CSTADeviceFilterMonitor CSTADeviceFilterRouteRegistrations, CSTADeviceFilterSubscriber, CSTADeviceFilterNID
DeviceSortType	Allows sorting by ServiceID-NID, EBGID, Monitored and Route Registration.	no	Enumeration	CSTADeviceSortTypeServiceId, CSTADeviceSortTypeEBGId CSTADeviceSortTypeMonitor CSTADeviceSortTypeRouteRegistration
SortOrder	Standard sorting structure			

Table 904 GetCSTAActiveDeviceListRequest Fields

Table 905 lists the fields of *GetCSTAActiveDeviceListResult*:

Field	Description	Type	Range - Length
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.		
PaginatedInfo	A common paging structure containing Current Page/ Num of Pages / Page Size etc.		
ActiveDeviceList	A list of active device elements		
(ActiveDeviceList) ActiveDeviceData	A structure containing the following data		
(ActiveDeviceData) DeviceId	For Subscriber Devices: If the application type is Telefonica then the Service ID is returned in EID+ExtensionFormat. For other application types the device number is presented in FQN format. For NID Devices: The NID of the device.	Char array	0..15
(ActiveDeviceData) EBGID	The EBGID. Applicable for Telefonica application type only and Subscriber device type.	Char array	1..32
(ActiveDeviceData) Monitored	Indicates whether this device has active monitors for the CSTA application.	Boolean	False, true
(ActiveDeviceData) RouteRegistration	Indicates whether this device has active Route Registrations for the CSTA application.	Boolean	False, true

Table 905 GetCSTAActiveDeviceListResult Fields

(ActiveDeviceData) Type	Indicates whether this device is a Subscriber or NID	Enumeration	CSTADeviceTypeSubscriber CSTADeviceTypeNID

Table 905 GetCSTAActiveDeviceListResult Fields

The following table shows the possible error codes of *GetCSTAActiveDeviceList*API:

WSDL Version	Return Codes	Message
V7.00.06.01	subRC_APP_COMM_OPERATOR_INVALID	Invalid Operator name. Length invalid or characters not allowed.
	subRC_APP_CSTA_APP_NOT_PRESENT	No CSTA Application with the specified IP Address exists
	subRC_APP_GENERAL_SDAL_ERROR	General SDAL Error

Table 906 GetCSTAActiveDeviceList Return Codes

21.12.6 DeleteCSTAApplication

The *DeleteCSTAApplication* API (starting with V7.00.06.01) will allow the deletion of a CSTA application.

Table 907 lists the fields of *DeleteCSTAApplicationRequest*:

Field	Description	Required	Type	Default Value [Range]
OperatorId	I/F operator	yes	Char array	NA [1…40]
ApplicationIPAddress	The (unique) IP of the application to be deleted in IPv4 format.	yes	Char array	1..45

Table 907 DeleteCSTAApplicationRequest Fields

Table 908 lists the fields of *DeleteCSTAApplicationResult*:

Field	Description	Type	Range - Length
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.		

Table 908 DeleteCSTAApplicationResult fields

The following table shows the possible error codes of *DeleteCSTAApplication* API:

WSDL Version	Return Codes	Message
V7.00.06.01	subRC_APP_COMM_OPERATOR_INVALID	Invalid Operator name. Length invalid or characters not allowed.
	subRC_APP_CSTA_APP_NOT_PRESENT	No CSTA Application with the specified IP Address exists
	subRC_APP_CSTA_APP_DEL_FAILED_MONITORED_DEVICES_EXIST	Failed to Delete CSTA Application. Active Monitored devices exist
	subRC_APP_CSTA_APP_DEL_FAILED_ROUTE_REG_DEVICES_EXIST	Failed to Delete CSTA Application. Route Registered devices exist
	subRC_APP_CSTA_APP_DEL_FAILED_CONNECTED	Failed to Delete CSTA Application. CSTA Application is connected
	subRC_APP_DB_GENERAL_SDAL_DB_ERROR	SDAL DB General Error
	subRC_APP_GENERAL_SDAL_ERROR	General SDAL Error

Table 909 DeleteCSTAApplication Return Codes

21.12.7 GetCSTASMInfo

The *GetCSTASMInfo* API (starting with V7.00.06.01) allows the *retrieval of the status of CSTA SM*.

Table 910 lists the fields of *GetCSTASMInfoRequest*:

Field	Description	Required	Type	Default Value [Range]
OperatorId	I/F operator	yes	Char array	NA [1…40]

Table 910 GetCSTASMInfoRequest Fields

Table 911 lists the fields of *GetCSTASMInfoResult*:

Field	Description	Type	Range - Length
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.		
CSTAStatus	A structure containing the following data		
(CSTAStatus) Status	The current status of CSTA SM.	Enumeration	CSTASMStatusNormal CSTASMStatusEnabled CSTASMStatusDisabled CSTASMStatusPartiallyDisabled CSTASMStatusOverloadReached CSTASMStatusOverloadRelieved
(CSTAStatus) LastUpdate	The date and time in text format of the last update of CSTA SM status.	Char array	0..256

Table 911 GetCSTASMInfo fields

22 CSTA Applications

SOAP supports the creation, modification, query and deletion of CSTA applications, the retrieval of the list of provisioned applications as well as the list of active devices for each application. When a CSTA application is created, modified or removed from the system, SOAP the associated packet filter rules are updated. Up to 256 applications are allowed to be provisioned. CSTA application can be provisioned using IPv4 addresses only.

The following APIs are used for the provisioning of CSTA applications:

- GetCSTAApplicationList
- CreateCSTAApplication
- ModifyCSTAApplication
- QueryCSTAApplication
- GetCSTAActiveDeviceList
- DeleteCSTAApplication
- GetCSTASMInfo

22.1 Get CSTA Application List

The *GetCSTAApplicationList* API (starting in V7.00.06.01) allows the retrieval of the list of CSTA applications provisioned in the system.

Table 912 lists the *input* fields for this method.

Field	Description	Required	Type	Default Value [Range]
OperatorId	I/F operator	yes	Char array	NA [1…40]
ApplicationType	Allows filtering of the result per application type or quering for all application	no	Enumerati o n	CSTAAppTypeAll CSTAAppTypeOpens cape CSTAAppTypeThirdP
Paging	Standard paging structure			
SortType	The sort criterion applied to the request.	no	Enumerati o n	CSTAAppSortTypeApplicatio nTyp e CSTAAppSortTypeIP CSTAAppSortTypeNameCS

Table 912 Get CSTA Application List Input Fields (Sheet 1 of 2)

Field	Description	Required	Type	Default Value [Range]
SortOrder	Standard sorting structure			

Table 912 Get CSTA Application List Input Fields (Sheet 2 of 2)

Table 913 lists the *output* fields for this method.

Field	Description	Type	Default Value [Range]
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.		
PaginatedInfo	A common paging structure containing Current Page/ Num of Pages / Page Size etc.		
ApplicationList	A list of ApplicationData elements		
(ApplicationList) ApplicationData	A structure containing the following data		
(ApplicationData) ApplicationName	Provides a description of the CSTA application. ApplicationName is a unique identifier. If a CSTA application is connected before it is provisioned it has a default name which is the application's IP address.	Char array	1..64
(ApplicationData) IPAddress	The (unique) IP of the application in IPv4 format.	Char array	1..45
(ApplicationData) ApplicationType	The type of the application. Value Unkown is set to an application if the application connects before provisioned.	Enumeration	CSTAAppTypeOpenscape CSTAAppTypeThidParty CSTAAppTypeTelefonica CSTAAppTypeUnknown
(ApplicationData) Connected	Indicates if the application is currently connected. This means that the application has established a CSTA over TCP or CSTA over HTTP/SOAP connection	Boolean	False, true
(ApplicationData) Monitors	The number of active CSTA monitored devices	Unsigned Integer	-
(ApplicationData) RouteRegistrations	The number of active CSTA route registration devices	Unsigned Integer	-
Totals	A structure containing the following data		

Table 913 Get CSTA Application List Output Fields (Sheet 1 of 2)

Field	Description	Type	Default Value [Range]
(Totals.) RouteRegistrations	The total number of active CSTA monitored devices	Unsigned Integer	-
(Totals.) Monitors	The total number of active CSTA monitored devices	Unsigned Integer	-

Table 913 Get CSTA Application List Output Fields (Sheet 2 of 2)

The following table shows the possible error codes of *GetCSTAApplicationList*

Table 914 lists the return codes for this method.

WSDL Version	Return Codes	
V7.00.06.01	subRC_APP_COMM_OPERATOR_INVALID	Invalid Operator name. Length invalid or characters not allowed.
	subRC_DB_GENERAL_SDAL_DB_ERROR	SDAL DB General Error

Table 914 Get CSTA Application List Return Codes

22.2 Create CSTA Application

The *CreateCSTAApplication* API (starting with V7.00.06.01) allows the provisioning of new CSTA applications. The creation of an application with existing IP address or name is blocked. Up to 256 applications are allowed to be created.

Table 915 lists the *input* fields for this method.

Field	Description	Required	Type	Default Value [Range]
OperatorId	I/F operator	yes	Char array	NA [1…40]
Identification	A required structure containing the following data			
(Identification.) ApplicationIPAddress	The (unique) IP of the application in IPv4 format. Maximum length is 45 characters	yes	Char array	1..45

Table 915 Create CSTA Application Input Fields (Sheet 1 of 4)

Field	Description	Required	Type	Default Value [Range]
(Identification.) ApplicationName	Provides a description of the CSTA application. The ApplicationName is a unique identifier of an application. If a CSTA application is connected before they are provisioned it has a default name which is the application's IP address. If the application type is Telefonica then this name is copied to the Application ID located in the Operational Data section.	yes	Char array	1..64
(Identification.) ApplicationType	CSTAAppTypeOpensc ape CSTAAppTypeThidPar ty CSTAAppTypeTelefoni ca	no	Enumerati o n	CSTAAppTypeOpenscap e (default) CSTAAppTypeThidParty CSTAAppTypeTelefonica
SOAPHeartbeat	A structure containing the following data.			
(SOAPHeartbeat) SOAPListenerPort	Port that applications use when connecting to OpenScape Voice via CSTA over HTTP/SOAP. CSTA over HTTP/SOAP uses same IP address as CSTA over TCP.	no	Unsigned Integer	Default: 7780 80, 1024 - 65535
(SOAPHeartbeat) HeartbeatTimer	System Status Heartbeat timer interval value. A value of 0 disables the heartbeat function.	no	Unsigned Integer	Default: 0 0-3600
(SOAPHeartbeat) HeartbeatWatchdog	System Status Heartbeat watchdog timer value.	no	Unsigned Integer	Default: 0 70..360
(SOAPHeartbeat) HeartbeatRetryCount	System Status Heartbeat retry value.	no	Unsigned Integer	Default: 0 0..5
OperationalData	A structure containing the following data. Applicable only to application of type ApplicationTypeTelefonica.			
(OperationalData) ApplicationID	The Application ID. If this field is not provided then it corresponds to the ApplicationName of the Identity section.	no	Char array	1..64
(OperationalData) EnterpriseID	Telefonica Enterprise Identification prefixed to subscriber deviceIDs presented to Telefonica application.	no	Char array	Default: 000000 6..8

Table 915 Create CSTA Application Input Fields (Sheet 2 of 4)

Field	Description	Required	Type	Default Value [Range]
(OperationalData) PASCODE	Telefonica prefixes this code to the routedSelected parameter when routing to the mobile network VPN. CRS uses this parameter to determine when to invoke Route End with RouteEndReason = endCstaDistribution.	no	Char array	Default: 000000 1..8
(OperationalData) OutsideAccess	Telefonica outside access code for external number conversion.	no	Char array	1..30
(OperationalData) BGName	Used to convert subscriber extension to fully qualified number via new mapping function.	no	Char array	1..30
(OperationalData) NPName	Used to convert subscriber extension to fully qualified number via new mapping function.	no	Char array	1..30
(Interface Options) EnableConsultCallForConnectOnHold	Indicates whether Consultation Call for Connections on Hold is enabled	no	Boolean	false [false, true]
(Interface Options) EnableCIDSuppressionStatus	Indicates whether CID Suppression Status is enabled.	no	Boolean	false [false, true]
(Interface Options) EnableAlternativeCIDList	Indicate whether Alternative CID List is enabled.	no	Boolean	false [false, true]
(Interface Options) DisableApplicationPopulatedCID	Indicates whether Application Populated CID is disabled.	no	Boolean	false [false, true]
(Interface Options) EnableCFPrivilegeChangeNotification	Indicates whether CSTA will get privilege update notifications	no	Boolean	(Interface Options) EnableCFPrivilegeChangeNotification
(Interface Options) EnableAllMembersBusyNotification	Indicates whether we get notification when all members of a Hunt Group are busy	no	Boolean	(Interface Options) EnableAllMembersBusyNotification
(Interface Options) Reserved_1	Reserved field	no	Boolean	false [false, true]
(Interface Options) Reserved_2	Reserved field	no	Boolean	false [false, true]

Table 915 Create CSTA Application Input Fields (Sheet 3 of 4)

Field	Description	Required	Type	Default Value [Range]
(Interface Options) Reserved_3	Reserved field	no	Boolean	false [false, true]

Table 915 Create CSTA Application Input Fields (Sheet 4 of 4)

Table 916 lists the *output* fields for this method.

Field	Description	Type	Default Value [Range]
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.		

Table 916 Create CSTA Application Output Fields

Table 917 lists the return codes for this method.

WSDL Version	Return Codes	Message
V7.00.06.01	subRC_APP_COMM_OPERATOR_INVALID	Invalid Operator name. Length invalid or characters not allowed.
	subRC_DB_GENERAL_SDAL_DB_ERROR	SDAL DB General Error
	subRC_APP_COMM_OPERATOR_INVALID	Invalid Operator name. Length invalid or characters not allowed.
	subRC_APP_CSTA_INVALID_VALUE	Invalid Value provided
	subRC_APP_CSTA_INVALID_APP_IP	Invalid Value provided
	subRC_APP_CSTA_INVALID_APP_TYPE	Invalid value provided: Application Type. Allowed values: Openscape, ThirdParty, Telefonica
	subRC_APP_CSTA_INVALID_APP_NAME	Invalid Value provided
	subRC_APP_CSTA_INVALID_OPERATION_DATA	Operational data may only be provided for Applications of type Telefonica Operational data are required for Applications of type Telefonica
	subRC_APP_CSTA_BG_NP_REQUIRED	At least one of BGName,NPName must be provided for Applications of type Telefonica
	subRC_APP_CSTA_APP_IP_EXISTS	A CSTA Application with the specified IP Address already exists
	subRC_APP_CSTA_APP_NAME_EXISTS	A CSTA Application with the specified Name already exists
	subRC_APP_CSTA_APP_MAX_APPS_REACHED	A CSTA Application with the specified Name already exists
	subRC_DB_GENERAL_SDAL_DB_ERROR	A CSTA Application with the specified Name already exists
	subRC_APP_GENERAL_SDAL_ERROR	A CSTA Application with the specified Name already exists
	subRC_APP_CSTA_BG_NP_MUTUALLY_EXCLUSIVE	BGName and NPName are mutually exclusive for Applications of type Telefonica
	subRC_APP_CSTA_PFR_UPDATE_ERROR	secPacketFilter.AddCstaAppRule Failed
	subRC_APP_CSTA_APP_SOASF_RTP_SEND_ERROR	Failed to send message to SOASF

Table 917 Create CSTA Application List Return Codes

22.3 Modify CSTA Application

The *ModifyCSTAApplication* API (starting with V7.00.06.01) allows the modification of a CSTA application. If the IP address of an application is changed SOAP will update the packets filtering rules. SOAP does not allow the modification of the ApplicationType of an existing application, except the transition from type *Unknown* to other types.

Table 918 lists the *input* fields for this method.

Field	Description	Required	Type	Default Value [Range]
OperatorId	I/F operator	yes	Char array	NA [1…40]
ApplicationIPAddress	The (unique) IP of the application to be updated in IPv4 format. Maximum length is 45 characters to allow future use of IPv6 addressing.	yes	Char array	1..45
Identification	A required structure containing the following data			
(Identification) ApplicationIPAddress	The new IP of the application in IPv4 format. This IP must unique among applications.	yes	Char array	1..45
(Identification) ApplicationName	Provides a description of the CSTA application. ApplicationName is a unique identifier. If a CSTA application is connected before they are provisioned it has a default name which is the application's IP address. If the application type is Telefonica then this name is copied to the Application ID located in the Operational Data section.	yes	Char array	1..64
(Identification) ApplicationType	The type of the application. Application type may be: CSTAAppTypeOpenscape (default) CSTAAppTypeThidParty CSTAAppTypeTelefonica	no	Enumeration	CSTAAppTypeOpenscape (default) CSTAAppTypeThidParty CSTAAppTypeTelefonica
SOAPHeartbeat	A structure containing the following data.			
(SOAPHeartbeat) SOAPListenerPort	Port that applications use when connecting to OpenScape Voice via CSTA over HTTP/ SOAP. CSTA over HTTP/ SOAP uses same IP address as CSTA over TCP.	no	Unsigned Integer	Default: 7780 80, 1024 - 65535

Table 918 Modify CSTA Application Input Fields (Sheet 1 of 3)

Field	Description	Required	Type	Default Value [Range]
(SOAPHeartbeat) HeartbeatTimer	System Status Heartbeat timer interval value. A value of 0 disables the heartbeat function.	no	Unsigned Integer	Default: 0 0-3600
(SOAPHeartbeat) HeartbeatWatchdog	System Status Heartbeat watchdog timer value.	no	Unsigned Integer	Default: 0 70..360
(SOAPHeartbeat) HeartbeatRetryCount	System Status Heartbeat retry value.	no	Unsigned Integer	Default: 0 0..5
OperationalData	A structure containing the following data. Applicable only to application of type ApplicationTypeTelefonica.			
(OperationalData) ApplicationID	The Application ID. If this field is not provided then it corresponds to the ApplicationName of the Identity section.	no	Char array	1..64
(OperationalData) EnterpriseID	Telefonica Enterprise Identification prefixed to subscriber deviceIDs presented to Telefonica application.	no	Char array	Default: 000000 6..8
(OperationalData) PASCODE	Telefonica prefixes this code to the routedSelected parameter when routing to the mobile network VPN. CRS uses this parameter to determine when to invoke Route End with RouteEndReason = endCstaDistribution.	no	Char array	Default: 000000 1..8
(OperationalData) OutsideAccess	Telefonica outside access code for external number conversion.	no	Char array	1..30
(OperationalData) BGName	Used to convert subscriber extension to fully qualified number via new mapping function.	no	Char array	1..30
(OperationalData) NPName	Used to convert subscriber extension to fully qualified number via new mapping function.	no	Char array	1..30
(Interface Options) EnableConsultCallForConnectOnHold	Indicates whether Consultation Call for Connections on Hold is enabled	no	Boolean	false [false, true]
(Interface Options) EnableCIDSuppressionStatus	Indicates whether CID Suppression Status is enabled.	no	Boolean	false [false, true]

Table 918 Modify CSTA Application Input Fields (Sheet 2 of 3)

Field	Description	Required	Type	Default Value [Range]
(Interface Options) EnableAlternativeCIDList	Indicate whether Alternative CID List is enabled.	no	Boolean	false [false, true]
(Interface Options) DisableApplicationPopulatedCID	Indicates whether Application Populated CID is disabled.	no	Boolean	false [false, true]
(Interface Options) EnableCFPrivilegeChangeNotification	Indicates whether CSTA will get privilege update notifications	no	Boolean	(Interface Options) EnableCFPrivilegeChangeNotification
(Interface Options) EnableAllMembersBusyNotification	Indicates whether we get notification when all members of a Hunt Group are busy	no	Boolean	(Interface Options) EnableAllMembersBusyNotification
(Interface Options) Reserved_1	Reserved field	no	Boolean	false [false, true]
(Interface Options) Reserved_2	Reserved field	no	Boolean	false [false, true]
(Interface Options) Reserved_3	Reserved field	no	Boolean	false [false, true]

Table 918 Modify CSTA Application Input Fields (Sheet 3 of 3)

Table 919 lists the *output* fields for this method.

Field	Description	Type	Default Value [Range]
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.		

Table 919 Modify CSTA Application List Output Fields

Table 920 lists the return codes for this method.

WSDL Version	Return Codes	
V7.00.06.01	subRC_APP_COMM_OPERATOR_INVALI D	Invalid Operator name. Length invalid or characters
	subRC_APP_CSTA_INVALID_VALUE	Invalid Value provided
	subRC_APP_CSTA_INVALID_APP_IP	Invalid Value provided
	subRC_APP_CSTA_INVALID_APP_TYPE	Invalid value provided: Application Type. Allowed
	subRC_APP_CSTA_INVALID_APP_NAME	Invalid Value provided
	subRC_APP_CSTA_INVALID_OPERATION _DATA	Operational data may only be provided for Applications of type Telefonica
	subRC_APP_CSTA_BG_NP_REQUIRED	At least one of BGName,NPName must be provided for
	subRC_APP_CSTA_APP_IP_EXISTS	A CSTA Application with the specified
	subRC_APP_CSTA_APP_NAME_EXISTS	A CSTA Application with the
	subRC_DB_GENERAL_SDAL_DB_ERRO R	A CSTA Application with the specified Name already exists
	subRC_APP_GENERAL_SDAL_ERROR	A CSTA Application with the
	subRC_APP_CSTA_BG_NP_MUTUALLY_ EXCLUSIVE	BGName and NPName are mutually exclusive for Applications of type
	subRC_APP_CSTA_PFR_UPDATE_ERRO R	secPacketFilter.ModifyCstaAppRule Failed
	subRC_APP_CSTA_APP_SOASF_RTP_S END_ERROR	Failed to send message to SOASF
	subRC_APP_CSTA_APP_NOT_PRESENT	No CSTA Application with the
	subRC_APP_CSTA_APP_MOD_FAILED_ MONITORED_DEVICES_EXIST	Failed to Modify CSTA Application. Active Monitored devices exist
	subRC_APP_CSTA_APP_MOD_FAILED_R OUTE_REG_DEVICES_EXIST	Failed to Modify CSTA Application. Route Registered devices exist
	subRC_APP_CSTA_APP_MOD_FAILED_CONNECTED	Failed to Modify CSTA Application. CSTA

Table 920 Modify CSTA Application List Return Codes

22.4 Query CSTA Application

The *QueryCSTAApplication* API (starting with V7.00.06.01) allows the retrieval of the data of a CSTA application provisioned in the system.

Table 921 lists the *input* fields for this method.

Field	Description	Required	Type	Default Value [Range]
OperatorId	I/F operator	yes	Char array	NA [1…40]
IPAddress	The (unique) IP of the application to be queried in IPv4 format.	yes	Char array	1..45

Table 921 Query CSTA Application Fields

Table 922 lists the *output* fields for this method.

Field	Description	Type	Default Value [Range]
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.		
Identification	A structure containing the following data		
(Identification) ApplicationIP	The IP of the application IPv4 format.	Char array	1..45
(Identification) ApplicationName	The name of the CSTA application.	Char array	1..64
(Identification) ApplicationType	The type of the application. Application type may be: ApplicationTypeOpenscape ApplicationTypeThidParty ApplicationTypeTelefonica ApplicationTypeUnkown	Enumeration	ApplicationTypeOpenscape ApplicationTypeThidParty ApplicationTypeTelefonica ApplicationTypeUnkown
SOAPHeartbeat	A structure containing the following data.		
(SOAPHeartbeat) SOAPListenerPort	Port that applications use when connecting to OpenScape Voice via CSTA over HTTP/SOAP. CSTA over HTTP/SOAP uses same IP address as CSTA over TCP.	Unsigned Integer	80, 1024 - 65535
(SOAPHeartbeat) HeartbeatTimer	System Status Heartbeat timer interval value. A value of 0 disables the heartbeat function.	Unsigned Integer	0-3600

Table 922 Query CSTA Application Output Fields (Sheet 1 of 4)

Field	Description	Type	Default Value [Range]
(SOAPHeartbeat.) HeartbeatWatchdog	System Status Heartbeat watchdog timer value.	Unsigned Integer	70..360
(SOAPHeartbeat.) HeartbeatRetryCount	System Status Heartbeat retry value.	Unsigned Integer	0..5
OperationalData	A structure containing the following data. Applicable only to application of type ApplicationTypeTelefonica.		
(OperationalData) ApplicationID	The Application ID. If this field is not provided then it corresponds to the ApplicationName of the Identity section.	Char array	1..64
(OperationalData) EnterpriseID	Telefonica Enterprise Identification prefixed to subscriber deviceIDs presented to Telefonica application.	Char array	6..8
(OperationalData) PASCODE	Telefonica prefixes this code to the routedSelected parameter when routing to the mobile network VPN. CRS uses this parameter to determine when to invoke Route End with RouteEndReason = endCstaDistribution.	Char array	1..8
(OperationalData) OutsideAccess	Telefonica outside access code for external number conversion.	Char array	1..30
(OperationalData) BGName	Used to convert subscriber extension to fully qualified number via new mapping function.	Char array	1..30
(OperationalData) NPName	Used to convert subscriber extension to fully qualified number via new mapping function.	Char array	1..30
ApplicationStatus	A structure containing the following data		
(ApplicationStatus) Status	The SOAP heartbeat status.	Enumeration	Active Inactive

Table 922 Query CSTA Application Output Fields (Sheet 2 of 4)

Field	Description	Type	Default Value [Range]
(ApplicationStatus) StatusLastUpdate	Date and time of last update of SOAP heartbeat status.	Char array	0..256
(ApplicationStatus) ConnectionStatus	The application connection status.	Enumeration	CSTAAppConnectionStatusDown CSTAAppConnectionStatusUp CSTAAppConnectionStatusFailed CSTAAppConnectionStatusRetrieved
(ApplicationStatus) ConnectionStatus LastUpdate	Date and time of last update of application connection status.	Char array	0..256
(ApplicationStatus) SessionStatus	The status of the application session service heartbeat.	Enumeration	Active Inactive
(ApplicationStatus) SessionStatus LastUpdate	Date and time of last update of the status of the application session service heartbeat.	Char array	0..256
SupportedWebService Names	A list of WebServiceName elements		
(SupportedWebService Names) WebServiceName	Supported web service name	Char array	1..64
(Interface Options) EnableConsultCallForConnectOnHold	Indicates whether Consultation Call for Connections on Hold is enabled	Boolean	false [false, true]
(Interface Options) EnableCIDSuppression Status	Indicates whether CID Suppression Status is enabled.	Boolean	false [false, true]
(Interface Options) EnableAlternativeCIDList	Indicate whether Alternative CID List is enabled.	Boolean	false [false, true]
(Interface Options) DisableApplicationPopulatedCID	Indicates whether Application Populated CID is disabled.	Boolean	false [false, true]
(Interface Options) Reserved_1	Reserved field	Boolean	false [false, true]

Table 922 Query CSTA Application Output Fields (Sheet 3 of 4)

Field	Description	Type	Default Value [Range]
(Interface Options) Reserved_2	Reserved field	Boolean	false [false, true]
(Interface Options) Reserved_3	Reserved field	Boolean	false [false, true]

Table 922 Query CSTA Application Output Fields (Sheet 4 of 4)

Table 923 lists the return codes for this method.

WSDL Version	Return Codes	
V7.00.06.01	subRC_APP_COMM_OPERATOR_INVALID	Invalid Operator name. Length invalid or characters not allowed.
	subRC_APP_CSTA_APP_NOT_PRESENT	No CSTA Application with the specified
	subRC_APP_GENERAL_SDAL_ERROR	General SDAL Error

Table 923 Query CSTA Application Return Codes

22.5 Get CSTA Active Device List

The GetCSTAActiveDeviceList API (starting with V7.00.06.01) allows the retrieval of the list of subscriber DNs with active CSTA monitors or route registrations for a CSTA application.

Table 924 lists the *input* fields for this method.

Field	Description	Required	Type	Default Value [Range]
OperatorId	I/F operator	yes	Char array	NA [1…40]
ApplicationIP	The (unique) IP of the application to be queried for active devices in IPv4 format.	yes	Char array	1..45
DeviceID	Allows filtering for a Service ID. Supports '*' and '?'	no	Char array	0..30
Paging	Standard paging structure			

Table 924 Get CSTA Active Device List Input Fields (Sheet 1 of 2)

Field	Description	Required	Type	Default Value [Range]
DeviceFilter	Allowes filtering for devices with active CSTA monitors, active CSTA route registrations, subscriber devices only, NID devices only or all devices.	no	Enumeration	CSTADeviceFilterAll CSTADeviceFilterMonitor CSTADeviceFilterRouteRegistrations, CSTADeviceFilterSubscriber,
DeviceSortType	Allows sorting by ServiceID-NID, EBGID, Monitored and Route Registration.	no	Enumeration	CSTADeviceSortTypeServiceId, CSTADeviceSortTypeEBGId CSTADeviceSortTypeMonitor
SortOrder	Standard sorting structure			

Table 924 Get CSTA Active Device List Input Fields (Sheet 2 of 2)

Table 925 lists the *output* fields for this method.

Field	Description	Type	Default Value [Range]
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.		
PaginatedInfo	A common paging structure containing Current Page/ Num of Pages / Page Size etc.		
ActiveDeviceList	A list of active device elements		
(ActiveDeviceList) ActiveDeviceData	A structure containing the following data		
(ActiveDeviceData) DeviceId	For Subscriber Devices: If the application type is Telefonica then the Service ID is returned in EID+ExtensionFormat. For other application types the device number is presented in FQN format. For NID Devices: The NID of the device.	Char array	0..15
(ActiveDeviceData) EBGID	The EBGID. Applicable for Telefonica application type only and Subscriber device type.	Char array	1..32

Table 925 Get CSTA Active Device List Output Fields (Sheet 1 of 2)

Field	Description	Type	Default Value [Range]
(ActiveDeviceData) Monitored	Indicates whether this device has active monitors for the CSTA application.	Boolean	False, true
(ActiveDeviceData) RouteRegistration	Indicates whether this device has active Route Registrations for the CSTA application.	Boolean	False, true
(ActiveDeviceData) Type	Indicates whether this device is a Subscriber or NID	Enumeration	CSTADeviceTypeSubscriber CSTADeviceTypeNID

Table 925 Get CSTA Active Device List Output Fields (Sheet 2 of 2)

Table 926 lists the return codes for this method.

WSDL Version	Return Codes	
V7.00.06.01	subRC_APP_COMM_OPERATOR_INVALID	Invalid Operator name. Length invalid or characters not allowed.
	subRC_APP_CSTA_APP_NOT_PRESENT	No CSTA Application with the specified
	subRC_APP_GENERAL_SDAL_ERROR	General SDAL Error

Table 926 Get CSTA Active Device List Return Codes

22.6 Delete CSTA Application

The *DeleteCSTAApplication* API (starting with V7.00.06.01) will allow the deletion of a CSTA application.

Table 927 lists the *input* fields for this method.

Field	Description	Required	Type	Default Value [Range]
OperatorId	I/F operator	yes	Char array	NA [1…40]
ApplicationIPAddress	The (unique) IP of the application to be deleted in	yes	Char array	1..45

Table 927 Delete CSTA Application Input Fields

Table 928 lists the *output* fields for this method.

Field	Description	Type	Default Value [Range]
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is		

Table 928 Delete CSTA Application Output Fields

Table 929 lists the return codes for this method.

WSDL Version	Return Codes	
V7.00.06.01	subRC_APP_COMM_OPERATOR_INVALID	Invalid Operator name. Length invalid or characters not allowed.
	subRC_APP_CSTA_APP_NOT_PRESENT	No CSTA Application with the specified IP Address exists
	subRC_APP_CSTA_APP_DEL_FAILED_MONITORED_DEVICES_EXIST	Failed to Delete CSTA Application. Active Monitored devices exist
	subRC_APP_CSTA_APP_DEL_FAILED_ROUTE_REG_DEVICES_EXIST	Failed to Delete CSTA Application. Route Registered devices exist
	subRC_APP_CSTA_APP_DEL_FAILED_CONNECTED	Failed to Delete CSTA Application. CSTA Application is connected
	subRC_APP_DB_GENERAL_SDAL_DB_ERROR	SDAL DB General Error
	subRC_APP_GENERAL_SDAL_ERROR	General SDAL Error

Table 929 Delete CSTA Application Return Codes

22.7 Get CSTASM Info

The *GetCSTASMInfo* API (starting with V7.00.06.01) allows the *retrieval of the status of CSTA SM*

Table 930 lists the *input* fields for this method.

Field	Description	Required	Type	Default Value [Range]
OperatorId	I/F operator	yes	Char array	NA [1…40]

Table 930 Get CSTASM Info Input Fields

Table 931 lists the *output* fields for this method.

Field	Description	Type	Default Value [Range]
Result	A typical structure containing return codes (indicating success/failure/error cases), invocation timestamp along with various informative messages. Return code list is displayed below.		
CSTAStatus	A structure containing the following data		
(CSTAStatus) Status	The current status of CSTA SM.	Enumeration	CSTASMStatusNormal CSTASMStatusEnabled CSTASMStatusDisabled CSTASMStatusPartiallyDisabled CSTASMStatusOverloadReached CSTASMStatusOverloadRelieved
(CSTAStatus) LastUpdate	The date and time in text format of the last update of CSTA SM status.	Char array	0..256

Table 931 Get CSTASM Info Output Fields

23 Alarms and Log Events

23.1 Alarms

The subscriber provisioning component issues alarms under some conditions, and in order to accomplish this, the following macro is called:

```
SRX_ALARM_EVENT ( eventSet,
                  eventId,
                  swnode,
                  faultyObject1,
                  faultyObject2,
                  faultyObject3,
                  errorText)
```

The following subsections describe each of the alarms that may be reported.

23.1.1 SOAP Server Process Exit

This alarm is issued when a SOAP Server child process ends for any reason.

This may be part of a normal shutdown of the OpenScape Voice system. If it is not, then subscriber management has experienced some problem during the processing of a subscriber provisioning request and either produced a core dump or exited in an attempt to restart its connections and recover.

This alarm will be reported each time soapServer child process terminates abnormally. The process will restart automatically. Since each SOAP Server child process uses a unique port number to accept SOAP requests, the port number is used in this alarm to identify which child process terminated. If the parent SOAP Server terminates abnormally, a number of alarms equal to the number of currently configured child processes (default: 4) will be reported.

The process is automatically restarted, and there is no manual action required by the operator other than to report the occurrence to OpenScape Voice support personnel. If the process ended due to a core dump, there may be symptoms such as a stack trace or a core dump file in the /unisphere/srx3000/srx/40/core directory.

Table 932 shows the values of the alarm event fields for this alarm.

Field	Description
Event Set ID	GLOBAL_EVENT_SET_ID (value: 217)
Event ID	GLOBAL_ERROR_EXIT_EVT_ID (value: 8)
Faulty Object 1	The node on which the alarm condition occurred.

Table 932 SOAP Server Process Exit Alarm Data (Sheet 1 of 2)

Field	Description
Faulty Object 2	“SrxSoapGroup”
Faulty Object 3	The listening port number of the SOAP Server child process that has terminated; for example “8767”.
Error Text	“Process terminating abnormally”

Table 932 SOAP Server Process Exit Alarm Data (Sheet 2 of 2)

23.1.2 SOAP Server Process Startup

This alarm is issued when a SOAP Server child process starts (or restarts after an exit).

Since each SOAP Server child process uses a unique port number to accept SOAP requests, the port number is used in this alarm to identify which child process started. On a normal startup of the SOAP Server, a number of alarms equal to the number of currently configured child processes (default: 4) will be reported.

This is a normal occurrence and there is no manual action required by the operator.

Note: This alarm clears the “SOAP Server Process Exit” alarm.

Table 933 shows the values of the alarm event fields for this alarm.

Field	Description
Event Set ID	GLOBAL_EVENT_SET_ID (value: 217)
Event ID	GLOBAL_PROCESS_ACTIVE_EVT_ID (value: 5)
Faulty Object 1	The node on which the alarm condition occurred.
Faulty Object 2	“SrxSoapGroup”
Faulty Object 3	The listening port number of the SOAP Server child process that has started up.
Error Text	“Process completely initialized and active”

Table 933 SOAP Server Process Startup Alarm Data

23.1.3 Submgt Schedule Process Exit

This alarm is issued when the submgtSchedule process ends for any reason.

This may be part of a normal shutdown of the OpenScape Voice system. If it is not, then subscriber management has experienced some problem during the processing of a background job and either produced a core dump or exited in an attempt to restart its connections and recover.

The process is automatically restarted, and there is no manual action required by the operator other than to report the occurrence to OpenScape Voice support personnel. If the process ended due to a core dump, there may be symptoms such as a stack trace or a core dump file in the /unisphere/srx3000/srx/40/core directory.

Table 934 shows the values of the alarm event fields for this alarm.

Field	Description
Event Set ID	GLOBAL_EVENT_SET_ID (value: 217)
Event ID	GLOBAL_ERROR_EXIT_EVT_ID (value: 8)
Faulty Object 1	The node on which the alarm condition occurred.
Faulty Object 2	“SrxSubmgtSchedGroup”
Faulty Object 3	“submgtSchedule<NN>”
Error Text	“Process terminating abnormally”

Table 934 Submgt Schedule Process Exit Alarm Data

Note: The “Faulty Object 3” field contains the exact name of the subscriber management scheduler process, which ends in “01” on one node of a cluster and “02” on the other. This information is redundant since the node is already specified in “Faulty Object 1” but is included for completeness.

23.1.4 Submgt Schedule Process Startup

This alarm is issued when the submgtSchedule process starts up (or restarts after an exit). This is a normal occurrence and there is no manual action required by the operator.

Note: This alarm clears the “SubmgtSchedule Process Exit” alarm.

Table 935 shows the values of the alarm event fields for this alarm.

Field	Description
Event Set ID	GLOBAL_EVENT_SET_ID (value: 217)
Event ID	GLOBAL_ERROR_EXIT_EVT_ID (value: 5)
Faulty Object 1	The node on which the alarm condition occurred.
Faulty Object 2	“SrxSubmgtSchedGroup”
Faulty Object 3	“submgtSchedule<NN>”
Error Text	“Process completely initialized and active”

Table 935 SubmgtSchedule Process Startup Alarm Data

Note: The “Faulty Object 3” field contains the exact name of the subscriber management scheduler process, which ends in “01” on one node of a cluster and “02” on the other. This information is redundant since the node is already specified in “Faulty Object 1” but is included for completeness.

23.1.5 Error Removing Old Jobs

This alarm is issued when the submgtSchedule process has failed to remove old jobs. These are background jobs that have been executed and remain stored in the database for a certain number of days after their completion (the exact number of days are configurable by RTP parameters, but by default the retention time is 3 days for successful jobs and 5 days for jobs that ended in error or were partially executed.

If the automatic removal of these jobs could not be performed for some reason, this alarm is reported.

There is no action necessary on the part of the operator other than to notify support personnel.

Table 936 shows the values of the alarm event fields for this alarm.

Field	Description
Event Set ID	SUBMGMT_EVENT_SET_ID (value: 207)
Event ID	SUBMGMT_REMOVE_RES_ERROR (value: 5)
Faulty Object 1	The node on which the alarm condition occurred.
Faulty Object 2	“SOAP”
Faulty Object 3	“JOBS”
Error Text	“The Job to remove old jobs failed: RC[<NN>]”

Table 936 Error Removing Old Jobs Alarm Data

Attention: <NN> in the “Error Text” field contains the error return code that was received from the lower level function call used by the submgtSchedule process to remove the old jobs.

23.1.6 Old Jobs Removed Successfully

This alarm is issued when the submgtSchedule process has successfully removed old jobs. These are background jobs that have been executed and remain stored in the database for a certain number of days after their completion

(the exact number of days are configurable by RTP parameters, but by default the retention time is 3 days for successful jobs and 5 days for jobs that ended in error or were partially executed.

Whenever the automatic removal of these jobs was performed successfully, this alarm is reported.

This is a normal occurrence and there is no action necessary on the part of the operator.

Note: This alarm clears the “Error Removing Old Jobs” alarm.

Table 937 shows the values of the alarm event fields for this alarm.

Field	Description
Event Set ID	SUBMGMT_EVENT_SET_ID (value: 207)
Event ID	SUBMGMT_REMOVE_RES_SUCCESS (value: 6)
Faulty Object 1	The node on which the alarm condition occurred.
Faulty Object 2	“SOAP”
Faulty Object 3	“JOBS”
Error Text	“Old resources removed”

Table 937 Old Jobs Removed Successfully Alarm Data

23.2 Log Events

The subscriber provisioning component issues log events under some conditions, and in order to accomplish this, the following macro is called:

```
OpLogReport        (fileName,
                   lineNumber,
                   logCategory,
                   logPriority,
                   logAppReason,
                   format,
                   args )
```

The following subsections describe each of the log events that may be reported. The fileName and lineNumber always indicate the location in the source code that originated the log event, and thus are not mentioned below. The “format” parameter allows for the possibility to report additional data by using a formatted string, and the contents of the “args” parameter correspond to the data items to be inserted into the formatted string.

23.2.1 Error Accessing the Database

This log event is issued whenever the processing of a SOAP request encounters a failure that corresponds to a database access error. Table 938 shows the values of the log event fields for this event.

Field	Description
logCategory	OP_LOG_CAT_DATABASE_CON_ERR
logPriority	OP_LOG_PRI_HIGH
logAppReason	0
format	“SOAP Database Access error, RC[%u]”
args	return code

Table 938 Example of Database Error

23.2.2 Provisioning Error

This log event is issued whenever the processing of a SOAP request encounters a failure that corresponds to an internal error. Table 939 shows the values of the log event fields for this event.

Field	Description
logCategory	OP_LOG_CAT_PROVISIONING_ERR
logPriority	OP_LOG_PRI_HIGH
logAppReason	0
format	“SOAP Provisioning error, RC[%u]”
args	return code

Table 939 Example of an Internal Error

23.3 Display of OSV Error Provisoning Logs

Effective starting with V6, ther is support of execution of log file exportation in an asynchronous way.

The Interfaces **unsp__CollectLogFiles** has been implemented that parses all log files: **HiQLogProvisioning.log.*** located under **/log** directory. The API is able identify log entries that their form is as the following example:

```
{
        serial_number => 44070,
        time => <Tue Sep 14 16:44:19 2010<,
        time_secs => 1284471859,
```

```
        time_usecs => 504918,
        process_name => <sipsm11<,
        file_name => <sipResolver.cc<,
        line_num => 319,
        pid => 10495,
        log_priority => <HIGH<,
        log_category => <SUB_PROVISIONING_ERR<,
        log_appl_reason => 72,
        component_id => <UNKNOWN<,
        appl_event_id => 72,
        num_prior_occurences => 0,
        managed_suppression_count => 10,
        managed_suppression_time => 1,
        data_length => 116,
        data => <SUBSCRIBE mesg received with invalid Group
        pickup URI access code=*78 from DN=302108189611
        device=172.25.12.180:5060<,
}
```

Table 940 lists the input fields for this method.

Field Name	Description	Length	Range	Default Value
Operator Id				
Log Category	Selection of errors from specific categories	Enum	SUB_PROVISIONING_ERR = 0, PROVISIONING_ERR	N/A
DateTime	User will be able to provide Date and Time so that log entries that have older timestamp are ignored. .	Char		

Table 940 Collect Log Files Input Fields

From each log entry the following rows will be exported

time => <Tue Sep 14 16:44:19 2010<,

log_category => <SUB_PROVISIONING_ERR<,

num_prior_occurences => 0,

data => <SUBSCRIBE mesg received with invalid Group pickup URI access code=*78 from DN=302108189611 device=172.25.12.180:5060<,

The first check is if exported value of **'time'** (Tue Sep 14 16:44:19 2010) is equal or greater than the time and date that user provided in the request. If this is the case, SOAP checks if the 'log_category' field coincides with the one that user provided. If positive, 'data' field is checked against a pool (vector) of previously stored log descriptions from previous log entry processing:

If contents of 'data' describe a new error, then SOAP saves the description of the error located inside 'data' field to the pool of error descriptions and creates a new record with the following elements:

Output: The output of the API is a file with tab separated fields showed below:

Field descriptions are located at the first line, tab separated. Field values are added to the following lines of the log file.

Output filename is the log category defined in the request.

- **OSV Node**: The IP address of the switch
- **Log Category**: Provided as input by the user, can be either **SUB_PROVISIONING_ERR** or **PROVISIONING_ERR**
- **Number of occurrences**:This field indicates the number of distinct records of this error found inside the log file. Note: It has to do nothing with num_prior_occurences. The latter may include prior occurrences of other error types.
- **Time&Date of first occurrence**: UPDATE with 'dateandtime' from the log entry
- **Time&Date of last occurrence**: UPDATE with 'dateandtime' from the log entry
- **Data** (error description): Copy the error description located at 'data' from the log entry. If contents of 'data' field match an existing description from the pool of descriptions of previous log entries, then SOAP locates this record and updates the following fields:

Table 941 lists the output fields for this method.

Field Name	Description	Length	Range	Default Value
Job Id	Job Id created by the Collect Log Files Request	Int		N/A
Log File Name	Log File Name that shall be created.	Char		

Table 941 Collect Log Files Input Fields

24 Import and Export of SOAP Provisioned Data

24.1 Export and Mass Provisioning

Subscriber Management Mass Provisioning allows the mass processing of provisioning commands by way of the CLI platform. To facilitate the provisioning of subscribers and their services, script files (containing shorthand versions of provisioning commands) are used. This section describes the format of the scripts and how to use them.

The Subscriber Management Mass Provisioning utility loops through the processing all of the commands in the script file. Output results are printed to the computer screen; or, the user may redirect the output to a file.

Warning: *soapMassProv* is intended to be used for performing changes *en masse.* Naturally such activities should only be scheduled during service windows or staging time. During normal call processing hours, provisioning should be limited to casual day to day activities through the SOAP admin interface on the OSV as the advised choice, for optimum load balancing reasons e.g. using CMP/OSV-Assistant or customer sponsored applications.

24.1.1 Importing Subscriber Data

Subscriber data can be imported to the OpenScape Voice system using the soapMassProv executable, which accepts a file in the same format as is generated by soapExport.

The executable is invoked from within the RTP CLI (startCLI) expert mode by using the command soapMassProv, followed by a string of parameters enclosed within double quotes. The syntax to perform this is as follows:

From the expert mode prompt:

CLI>soapMassProv “-f=*ProvisioningCommandFile*”

where

ProvisioningCommandFile is a file containing the commands to execute. Refer to Section B.3, “Command Line Syntax”, on page 978 for a detailed description of the syntax of this file.

It is also possible to execute soapMassProv from the command line, as was supported in the past, but now this method is reserved for internal development and test purposes only, and therefore the operator ID and password parameters are required in addition to the parameters listed above. The command line syntax is as follows:

1. ssh to the OpenScape Voice system as user “srx”.
2. cd ../UNSPsubp
3. ./soapMassProv -o=*Operator* -pass=*Password* -f=*ProvisioningCommandFile*

Table 942 lists and describes the soapMassProv parameters.

Parameter	Type	Description
-f=filename	Command line parameter	Specifies the input file that contains mass provisioning commands to be processed. The valid syntax for this file is defined in the mp2.dat file and also later in this document.
-h=homeDirectory	Command line parameter	Specifies a directory that is different from the one from which the program is started. Used by the program to find configuration information. UNSPsubp is installed in /Unisphere/srx3000/UNSPsubp and it should be executed from there If specified from another directory, specify -h=/Unisphere/srx3000/UNSPsubp.
-out=outputFile	Command line parameter	Specifies an output file to which the following data is written: For successfully processed commands, the command from the import file is copied here, but commented out. For commands that failed for any reason, the command from the input file is copied here without comments. This means that, for example, in case of sequence problems, where some commands must be executed as a prerequisite for other commands, the output file can be reused as an input file for a second pass to complete the processing of those commands that failed the first time.
-p=priority	Command line parameter	Specifies the priority at which this process it to be run, ranging from 0 to -60, where -60 is the default if the parameter is not provided.
-o=operatorId	Command line parameter	A string identifying the operator who is performing this action; only required when performing this action from the command line as opposed to via expert mode CLI.
-pass=password	Command line parameter	The password that is required in order to perform this action from the command line as opposed to via expert mode CLI.

Table 942 soapMassProv Parameters (Sheet 1 of 2)

Parameter	Type	Description
-lins	Command line parameter	-lins=<lin list size> : exports defined lin list size -lins : exports all lins from Bg Subnet
-Mlhg	Command line parameter	-Mlhg=<DN> : exports Multiline Hunt Group <DN> -Mlhg : Exports ALL Multiline Hunt Groups
-LOT=<lotname>	Command line parameter	Exports <lotname> from the defined BG. -lins : exports all parking lots from the specified BG.
-EAC:BG=BGName	Command line parameter	BGname specifies a business group name. The result is a list of E/A Groups, members, relations and AuxLines.
-MSA=BGname:BOname	Command line parameter	BGname specifies a business group name, BOname specifies the branch office name. The result is a list of media server assignments to the specified branch office.
-passupdate	Command line parameter	Will prompt for the current password and give the option to update the mass provisioning password.

Table 942 soapMassProv Parameters (Sheet 2 of 2)

If -StartDn, -c and -Mlhg are specified, all defined subscribers and multiline hunt groups inside this range are exported.

If -BG and -lins are specified, all lins from a BG subnet are exported.

If both -BG and -LOT are specified, all parking lots are exported. -LOT exports all parking lots, and -LOT=<lotname> exports a specific parking lot.

Usage Example for -passupdate

```
./soapExport -o=user -f=output-file -BG -passupdate
Enter current password to proceed:
Authentication OK
---Update password---
Type new password:
*****
Type new password:
*****
Password updated successfully!
```

24.1.2 Exporting Subscriber Data

Subscriber data can be exported from the OpenScape Voice system to a comma delimited file.

The executable is invoked from within the RTP CLI (startCLI) expert mode by using the command soapExport, followed by a string of parameters enclosed within double quotes. The syntax to perform this is as follows:

From the expert mode prompt:

CLI>soapExport “-f=*outputfilename* [-BG]-StartDN=*StartDn* [-c=*count*]]”

It is also possible to execute soapMassProv from the command line, as was supported in the past, but now this method is reserved for internal development and test purposes only, and therefore the operator ID and password parameters are required in addition to the parameters listed above. The command line syntax is as follows:

1. ssh to the OpenScape Voice system as user “srx”.
2. cd ../UNSPsubp
3. ./soapExport -o=*Operator* -pass=*Password* -f=*outputfilename* [-BG]-StartDN=*StartDn* [-c=*count*]]

where *outputfilename* is the location to which the exported data is to be written.

BG indicates a specific BGName or all business group definitions

PNP specifies a NumberPlanName or the entire Private Numbering Plan

EP specifies an EndPointProfile or all End Point Profiles

QP specifies a QOS Profile or all QOS Profiles

FP specifies a Feature Profile or all Feature Profiles

StartDn is the subscriber to start with

NM specifies all the features of the Network Management interfaces

TS specifies the Tombstone table

count is the max number of subscribers to export

- If **-BG** and **-PNP** are specified, the Business Group will be exported with its own Private Numbering Plan.
- If **-BG** and **-StartDn** are specified, the Business Group will be exported with all its subscribers starting from the specified DN.
- If neither **-BG**, **-PNP**, or **-StartDn** are specified, the subscribers will be exported starting with the lowest numerical alphabetic.
- If **-NM=All** is specified, -NM=<ObjectName> will export only that object. For example,
  - soapExport "-NM=All -f=/tmp/nm":// All objects
  - soapExport "-NM=OfficeCode -f=/tmp/nm :// Exports OfficeCodes

The output from *soapExport* can be used as input to *soapMassProv*.

Table 943 lists and describes the soapExport parameters.

Parameter	Type	Description
-f=filename	Command line parameter	Specifies the output file that will be generated with the comma-delimited data.

Table 943 soapExport Parameters (Sheet 1 of 3)

Parameter	Type	Description
-o=operatorId	Command line parameter	A string identifying the operator who is performing this action; only required when performing this action from the command line as opposed to via expert mode CLI.
-pass=password	Command line parameter	The password that is required in order to perform this action from the command line as opposed to via expert mode CLI.
-BG=BGName	Command line parameter	Specifies a Business Group Name. If the parameter is provided with no value, all Business Group definitions are exported.
-EP = EPName	Command line parameter	Specifies an Endpoint Profile. If the parameter is specified with no value, all Endpoint Profiles are exported.
-FP = Feature Profile Name	Command line parameter	Specifies a Feature Profile. If the parameter is provided with no value, all Feature Profiles are exported. Since Feature Profiles may or may not be associated with Business Groups, the following other possibilities apply: • FP --- export all Feature Profiles • FP=: --- export all Switch Wide Feature Profiles • FP=<fp> --- export Feature Profile with Feature Profile Name = <fp> • FP=:<bg> --- export all Feature Profiles • belonging to Business Group <bg> • FP=<fp>:<bg> --- export Feature Profile with Feature Profile Name <fp> belonging to Business Group <bg> Note this means that if a Feature Profile is associated with a BG, the BG Name must be provided.
-PNP=NumberPlan Name	Command line parameter	Specifies a NumberPlanName. If not specified, the BG must be specified, and the Private Numbering Plan associated with that BG is exported.
-DEST	Command line parameter	When included along with the -BG and -PNP options, indicates that all destinations for the given Numbering Plan are to be exported.
-NumMod=Number PlanName	Command line parameter	Specifies a NumberPlanName for which the Number Modification data is to be exported.
-TS	Command line parameter	Indicates that the contents of the Tombstone table are to be exported.
-StartDn=DN	Command line parameter	Indicates the starting Directory Number at which to start the export. If not specified, export starts with the numerically lowest DN provisioned on the system.
-c=Number	Command line parameter	Indicates the number of Directory Numbers to export. If not specified, or greater than 10,000 is specified, 10,000 will be exported.

Table 943 soapExport Parameters (Sheet 2 of 3)

Parameter	Type	Description
-c2=Number	Command line parameter	Same as for the -c option, this parameter indicates the number of Directory Numbers to export; however, in this case, there is no limit to the number of exported DNs. Note that if the -c2 option is used on a heavily populated system, the result may be a long running command.
-NM=All -NM=<object>	Command line parameter	Indicates that Network Management objects are to be exported. If gAllh is specified, all supported objects are exported. Alternatively, an object name can be provided to export only that object. The currently supported Network Management objects are: • OfficeCode • HomeDn • ClassService • RateArea • CallingLocation • Destination • Endpoint • Route • Alias • MediaGateway • MediaGatewayCircuits • TimeDestination • DaySchedule • PeriodSchedule • WeeklySchedule • Intercept • Treatment • CarrierDestination • CarrierRoute • TrunkGroup • Trunk • OriginDestination • OriginRoute • SipReason • CodeProcess • OperatorEnum • EnumServer • EndpointProvider • EndpointLocation
-lins	Command line parameter	-lins=<lin list size> : exports defined lin list size -lins : exports all lins from Bg Subnet
-Mlhg	Command line parameter	-Mlhg=<DN> : exports Multiline Hunt Group <DN> -Mlhg : Exports ALL Multiline Hunt Groups

Table 943 soapExport Parameters (Sheet 3 of 3)

If -StartDn, -c and -Mlhg are specified, all defined subscribers and multiline hunt groups inside this range are exported.

If -BG and -lins are specified all lins from a BG subnet are exported.

The soapExport exports the provisioning commands in the current version and outputs them to the file. This file can then be imported (using soapMassProv) with a version greater than or equal to the version output in the file.

Beginning in Version 3.0, the SOAP Export and SOAP Mass Provisioning syntax is enhanced to improve its readability. Using the old style format was cumbersome due to the fact that the parameters were position-dependent and thus the mapping of data items to parameters was difficult to determine by examining the script file. Therefore, a new style of command is introduced consisting of a parameter = value format, making the parameters clearly identifiable and position independent.

Every command has the general format:

```
<command-name>
 ,<attribute1>=<value1>
 ,<attribute2>=<value2>
 ,<attribute3>=<value3>
..
 ,<attribute4>=<value4>
;;
```

Warning: Note one important restriction of this new format: The command and attribute names are reserved words. This means that if they are to be used as an input value for some attribute, they must be placed within double quotes. For example, to create a Business Group with the name “AF”, which is a reserved command name, the following command could be used:

```
CBG
,BGNAME=”AF”
,DISPLAYNUMBER=5619556559
```

Also note the following about SOAP Mass Provisioning:

- Files that have a file version with an “MP2” are parsed with the new lex-yacc parser.
- C style comment syntax has been introduced. This allows large blocks of input file to be commented out if necessary with /* and */.

For your reference and convenience, the classic (old style) SOAP Export and SOAP Mass Provisioning syntax is documented in Appendix B, “Classic Import and Export Information”.

24.2 Supported Commands

The following subsections present the syntax for each of the supported commands. This information is also included in the deliverable file **mp2.dat**. For specific details about attributes, please refer to this file. Additionally, the files

generated by SOAP Export now have the WSDL version included at the top of the file in order to clearly document the version of the switch from which the data was exported.

24.2.1 Add Features

```
<AF>
AF - Add Features to subscribers (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
   ,SUBSCRIBERDN=<string>
   feat1
,feat1_attribute1=<value>
,feat1_attribute2=<value>
   feat2
,feat2_attribute1=<value>
,feat2_attribute2=<value>
   .
   .
   featn
,featn_attribute1=<value>
,featn_attribute2=<value>
</AF>
```

24.2.1.1 Supported Features

Following is a list of features supported by the Add Subscriber Feature (AF) command above along with their attributes. Note that in general, these features are also supported by the other subscriber feature listed below:

- MF - Modify Subscriber Feature
- IF - Reset Inherited Subscriber feature fields
- FPAF - Add Feature Profile Feature
- FPRF - Remove Feature Profile Feature
- FPMF - Modify Feature Profile Feature
- FPMFR - Modify Feature Profile Feature with Reset of Subs

```
<CFACCB>
CFACCB,
ENABLED=<bool>
</CFACCB>

<CFACR>
ACR -  (Please see WSDL API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
,SUBCONTROLLED=<bool>
,BILLING=<enum>
</CFACR>
```

```
<CFAcctCode>
AcctCode -  (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
</CFAcctCode>

<CfAuthCode>
AuthCode -  (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
,AUTHCODECDR=<enum>
,INTERNATIONAL=<bool>
,INTERNATIONALZONE1=<bool>
,INTERLATA=<bool>
,INTRALATA=<bool>
,TOLLFREE=<bool>
,LOCALDIRASSIST=<bool>
,HOMEDIRASSIST=<bool>
,LONGDIRASSIST=<bool>
,EMERGENCY=<bool>
,NATIONAL=<bool>
,LOCALTT=<bool>
</CfAuthCode>

<CFBgCNAM>
BgCNAM -  (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
</CFBgCNAM>

<CFBgCPU>
BgCPU -  (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
</CFBgCPU>

<CFCPUD>
CPUD - Call Pick Up Direct (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
</CFCPUD>

<CFBgCT>
BgCT -  (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,BGCTVARIANT=<enum>
,BGCTRECALLAPPLYTYPE=<enum>
,BGCTRECALLTIMER=<int>
,BGCTINTERCEPTAPPLYTYPE=<enum>
,BGCTINTERCEPTTIMER=<int>
,BGCTINTERCEPTDEST=<string>
,BGCTBLOCKTRANSFERTORESTRICTEDPARTY=<bool>
</CFBgCT>

<CFBgDAPPS>
BgDAPPS -  (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,INTRABGCALLSPPS=<enum>
,CALLINGINTRABGDISPLAYFIRSTOPTION=<enum>
,CALLINGINTRABGDISPLAYSECONDOPTION=<enum>
,CALLINGINTRABGDISPLAYTHIRDOPTION=<enum>
,CALLINGNONINTRABGDISPLAYFIRSTOPTION=<enum>
```

```
,CALLINGNONINTRABGDISPLAYSECONDOPTION=<enum>
,CONNECTEDINTRABGDISPLAYFIRSTOPTION=<enum>
,CONNECTEDINTRABGDISPLAYSECONDOPTION=<enum>
,CONNECTEDINTRABGDISPLAYTHIRDOPTION=<enum>
</CFBgDAPPS>

<CFBgDNPPS>
BgDNPPS -  (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,DEFAULTPPS=<enum>
,INTRABGCALLSPPS=<enum>
,EXTERNALCALLERID=<string>
</CFBgDNPPS>

<CFCACC>
CACC - "Cancel Automatic Collect Call" (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
</CFCACC>

<CFCCBS>
CCBS - Completion of Calls on Busy Subscriber service
(Please see WSDL API:unsp__UpdateSubscriberFeatures)
</CFCCBS>

<CFCCNR>
CCNR - Completion of Calls on No Reply service (Please see
WSDL API:unsp__UpdateSubscriberFeatures)
</CFCCNR>

<CFCFB>
CFB - Call Forward Busy (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
,SERVICEID=<string>
,SUBACTIVATABLE=<enum>
,DESTSERVICEIDSUBCONTROLLABLE=<enum>
,CFNOTIFYCALLINGPARTY=<enum>
,NOTIFYSUBCALLFWDACTIVE=<bool>
</CFCFB>

<CFCFD>
CFCFD - Call Forward Dependable (Please see WSDL
API:unsp__UpdateSubscriberFeatures in Provisioning Doc for
Attribute details)
,ENABLED=<bool>
,SERVICEID=<string>
,EXTSERVICEID=<string>
</CFCFD>

<CFCFDA>
CFDA - Call Forward No Answer (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
,SERVICEID=<string>
,NUMBEROFRINGS=<int>
,RINGDURATION=<int>
,SUBACTIVATABLE=<enum>
,DESTSERVICEIDSUBCONTROLLABLE=<enum>
,CFNOTIFYCALLINGPARTY=<enum>
,NOTIFYSUBCALLFWDACTIVE=<bool>
</CFCFDA>
```

```
<CFCFV>
CFV - Call Forward Variable (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
,SERVICEID=<string>
,SUBACTIVATABLE=<enum>
,DESTSERVICEIDSUBCONTROLLABLE=<enum>
,CFNOTIFYCALLINGPARTY=<enum>
,NOTIFYSUBCALLFWDOCCURRED=<bool>
,NOTIFYSUBCALLFWDACTIVE=<bool>
</CFCFV>

<CFCIDS>
CIDS - Outgoing Caller ID Suppression (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
</CFCIDS>

<CFCPS>
CFCPS - Call Park to Server (Please see WSDL
API:unsp__UpdateSubscriberFeatures for attribute details)
 , SUBSCRIBERDN=<string>
, ENABLED=<bool>
, PARKALLOW=<bool>
, RETRIEVEALLOW=<bool>
, LOTNAME=<string>
, BGNAME=<string>
</CFCPS>

<CFCPUD>
CPUD - Call Pick Up Direct (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
</CFCPUD>

<CFCRBT>
CFCRBT - Colorful Ringback Tone (Please see WSDL
API:unsp__UpdateSubscriberFeatures in Provisioning Doc for
Attribute details)
,ENABLED=<bool>
,ACTIVERINGBACKTONE=<string>
,RBTONELIST_LEN=<int>
,INTERCEPTNAME_0=<string>
,INTERCEPTNAME_1=<string>
,INTERCEPTNAME_2=<string>
,INTERCEPTNAME_3=<string>
,INTERCEPTNAME_4=<string>
,INTERCEPTNAME_5=<string>
,INTERCEPTNAME_6=<string>
,INTERCEPTNAME_7=<string>
,INTERCEPTNAME_8=<string>
,INTERCEPTNAME_9=<string>
</CFCRBT>

<CFCT>
CT - Call Transfer (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
</CFCT>

<CFCSTA>
CSTA - "CSTA for SimplyPhone" (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
```

```
,CSTATYPE=<enum>
</CFCSTA>

<CFCW>
CW -  (Please see WSDL API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
,NOTIFYCALLINGPARTY=<bool>
</CFCW>

<CFCFDISP>
CFDISP - Call Forward Display (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
</CFCFDISP>

<CFDND>
DND -  (Please see WSDL API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
</CFDND>

<CFDTF>
DTF - Deny Terminating Feature (Please see WSDL
API:unsp__UpdateSubscriberFeatures in Provisioning Doc for
Attribute details)
,ENABLED=<bool>
,CALLBACKMONITORING=<bool>
</CFDTF>

<CFECF>
ECF - "Enhanced Call forward" (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
,SCREENLIST_LEN=<int>
,SCREENLIST_SERVICEID_0=<string>
,SCREENLIST_ENTRY_0=<enum>
,SCREENLIST_SERVICEID_1=<string>
,SCREENLIST_ENTRY_1=<enum>
,SCREENLIST_SERVICEID_2=<string>
,SCREENLIST_ENTRY_2=<enum>
,SCREENLIST_SERVICEID_3=<string>
,SCREENLIST_ENTRY_3=<enum>
,SCREENLIST_SERVICEID_4=<string>
,SCREENLIST_ENTRY_4=<enum>
,SCREENLIST_SERVICEID_5=<string>
,SCREENLIST_ENTRY_5=<enum>
,SCREENLIST_SERVICEID_6=<string>
,SCREENLIST_ENTRY_6=<enum>
,SCREENLIST_SERVICEID_7=<string>
,SCREENLIST_ENTRY_7=<enum>
,SCREENLIST_SERVICEID_8=<string>
,SCREENLIST_ENTRY_8=<enum>
,SCREENLIST_SERVICEID_9=<string>
,SCREENLIST_ENTRY_9=<enum>
,SCREENLIST_SERVICEID_10=<string>
,SCREENLIST_ENTRY_10=<enum>
,SCREENLIST_SERVICEID_11=<string>
,SCREENLIST_ENTRY_11=<enum>
,SCREENLIST_SERVICEID_12=<string>
,SCREENLIST_ENTRY_12=<enum>
,SCREENLIST_SERVICEID_13=<string>
,SCREENLIST_ENTRY_13=<enum>
,SCREENLIST_SERVICEID_14=<string>
```

```
,SCREENLIST_ENTRY_14=<enum>
,SCREENLIST_SERVICEID_15=<string>
,SCREENLIST_ENTRY_15=<enum>
,SCREENLIST_SERVICEID_16=<string>
,SCREENLIST_ENTRY_16=<enum>
,SCREENLIST_SERVICEID_17=<string>
,SCREENLIST_ENTRY_17=<enum>
,SCREENLIST_SERVICEID_18=<string>
,SCREENLIST_ENTRY_18=<enum>
,SCREENLIST_SERVICEID_19=<string>
,SCREENLIST_ENTRY_19=<enum>
,SCREENLIST_SERVICEID_20=<string>
,SCREENLIST_ENTRY_20=<enum>
,SCREENLIST_SERVICEID_21=<string>
,SCREENLIST_ENTRY_21=<enum>
,SCREENLIST_SERVICEID_22=<string>
,SCREENLIST_ENTRY_22=<enum>
,SCREENLIST_SERVICEID_23=<string>
,SCREENLIST_ENTRY_23=<enum>
,SCREENLIST_SERVICEID_24=<string>
,SCREENLIST_ENTRY_24=<enum>
,SCREENLIST_SERVICEID_25=<string>
,SCREENLIST_ENTRY_25=<enum>
,SCREENLIST_SERVICEID_26=<string>
,SCREENLIST_ENTRY_26=<enum>
,SCREENLIST_SERVICEID_27=<string>
,SCREENLIST_ENTRY_27=<enum>
,SCREENLIST_SERVICEID_28=<string>
,SCREENLIST_ENTRY_28=<enum>
,SCREENLIST_SERVICEID_29=<string>
,SCREENLIST_ENTRY_29=<enum>
,SCREENLIST_SERVICEID_30=<string>
,SCREENLIST_ENTRY_30=<enum>
,SCREENLIST_SERVICEID_31=<string>
,SCREENLIST_ENTRY_31=<enum>
,ECFLIST_LEN=<int>
,ECFLIST_DOW_0=<enum>
,ECFLIST_STARTTIME_0=<string>
,ECFLIST_STOPTIME_0=<string>
,ECFLIST_CALLFWDTYPE_0=<enum>
,ECFLIST_SCREENOPTION_0=<enum>
,ECFLIST_SERVICEID_0=<string>
,ECFLIST_DOW_1=<enum>
,ECFLIST_STARTTIME_1=<string>
,ECFLIST_STOPTIME_1=<string>
,ECFLIST_CALLFWDTYPE_1=<enum>
,ECFLIST_SCREENOPTION_1=<enum>
,ECFLIST_SERVICEID_1=<string>
,ECFLIST_DOW_2=<enum>
,ECFLIST_STARTTIME_2=<string>
,ECFLIST_STOPTIME_2=<string>
,ECFLIST_CALLFWDTYPE_2=<enum>
,ECFLIST_SCREENOPTION_2=<enum>
,ECFLIST_SERVICEID_2=<string>
,ECFLIST_DOW_3=<enum>
,ECFLIST_STARTTIME_3=<string>
,ECFLIST_STOPTIME_3=<string>
,ECFLIST_CALLFWDTYPE_3=<enum>
,ECFLIST_SCREENOPTION_3=<enum>
,ECFLIST_SERVICEID_3=<string>
,ECFLIST_DOW_4=<enum>
,ECFLIST_STARTTIME_4=<string>
```

```
,ECFLIST_STOPTIME_4=<string>
,ECFLIST_CALLFWDTYPE_4=<enum>
,ECFLIST_SCREENOPTION_4=<enum>
,ECFLIST_SERVICEID_4=<string>
,ECFLIST_DOW_5=<enum>
,ECFLIST_STARTTIME_5=<string>
,ECFLIST_STOPTIME_5=<string>
,ECFLIST_CALLFWDTYPE_5=<enum>
,ECFLIST_SCREENOPTION_5=<enum>
,ECFLIST_SERVICEID_5=<string>
,ECFLIST_DOW_6=<enum>
,ECFLIST_STARTTIME_6=<string>
,ECFLIST_STOPTIME_6=<string>
,ECFLIST_CALLFWDTYPE_6=<enum>
,ECFLIST_SCREENOPTION_6=<enum>
,ECFLIST_SERVICEID_6=<string>
,ECFLIST_DOW_7=<enum>
,ECFLIST_STARTTIME_7=<string>
,ECFLIST_STOPTIME_7=<string>
,ECFLIST_CALLFWDTYPE_7=<enum>
,ECFLIST_SCREENOPTION_7=<enum>
,ECFLIST_SERVICEID_7=<string>
,ECFLIST_DOW_8=<enum>
,ECFLIST_STARTTIME_8=<string>
,ECFLIST_STOPTIME_8=<string>
,ECFLIST_CALLFWDTYPE_8=<enum>
,ECFLIST_SCREENOPTION_8=<enum>
,ECFLIST_SERVICEID_8=<string>
,ECFLIST_DOW_9=<enum>
,ECFLIST_STARTTIME_9=<string>
,ECFLIST_STOPTIME_9=<string>
,ECFLIST_CALLFWDTYPE_9=<enum>
,ECFLIST_SCREENOPTION_9=<enum>
,ECFLIST_SERVICEID_9=<string>
,ECFLIST_DOW_10=<enum>
,ECFLIST_STARTTIME_10=<string>
,ECFLIST_STOPTIME_10=<string>
,ECFLIST_CALLFWDTYPE_10=<enum>
,ECFLIST_SCREENOPTION_10=<enum>
,ECFLIST_SERVICEID_10=<string>
,ECFLIST_DOW_11=<enum>
,ECFLIST_STARTTIME_11=<string>
,ECFLIST_STOPTIME_11=<string>
,ECFLIST_CALLFWDTYPE_11=<enum>
,ECFLIST_SCREENOPTION_11=<enum>
,ECFLIST_SERVICEID_11=<string>
,ECFLIST_DOW_12=<enum>
,ECFLIST_STARTTIME_12=<string>
,ECFLIST_STOPTIME_12=<string>
,ECFLIST_CALLFWDTYPE_12=<enum>
,ECFLIST_SCREENOPTION_12=<enum>
,ECFLIST_SERVICEID_12=<string>
,ECFLIST_DOW_13=<enum>
,ECFLIST_STARTTIME_13=<string>
,ECFLIST_STOPTIME_13=<string>
,ECFLIST_CALLFWDTYPE_13=<enum>
,ECFLIST_SCREENOPTION_13=<enum>
,ECFLIST_SERVICEID_13=<string>
,ECFLIST_DOW_14=<enum>
,ECFLIST_STARTTIME_14=<string>
,ECFLIST_STOPTIME_14=<string>
,ECFLIST_CALLFWDTYPE_14=<enum>
```

```
,ECFLIST_SCREENOPTION_14=<enum>
,ECFLIST_SERVICEID_14=<string>
,ECFLIST_DOW_15=<enum>
,ECFLIST_STARTTIME_15=<string>
,ECFLIST_STOPTIME_15=<string>
,ECFLIST_CALLFWDTYPE_15=<enum>
,ECFLIST_SCREENOPTION_15=<enum>
,ECFLIST_SERVICEID_15=<string>
,ECFLIST_DOW_16=<enum>
,ECFLIST_STARTTIME_16=<string>
,ECFLIST_STOPTIME_16=<string>
,ECFLIST_CALLFWDTYPE_16=<enum>
,ECFLIST_SCREENOPTION_16=<enum>
,ECFLIST_SERVICEID_16=<string>
,ECFLIST_DOW_17=<enum>
,ECFLIST_STARTTIME_17=<string>
,ECFLIST_STOPTIME_17=<string>
,ECFLIST_CALLFWDTYPE_17=<enum>
,ECFLIST_SCREENOPTION_17=<enum>
,ECFLIST_SERVICEID_17=<string>
,ECFLIST_DOW_18=<enum>
,ECFLIST_STARTTIME_18=<string>
,ECFLIST_STOPTIME_18=<string>
,ECFLIST_CALLFWDTYPE_18=<enum>
,ECFLIST_SCREENOPTION_18=<enum>
,ECFLIST_SERVICEID_18=<string>
,ECFLIST_DOW_19=<enum>
,ECFLIST_STARTTIME_19=<string>
,ECFLIST_STOPTIME_19=<string>
,ECFLIST_CALLFWDTYPE_19=<enum>
,ECFLIST_SCREENOPTION_19=<enum>
,ECFLIST_SERVICEID_19=<string>
,ECFLIST_DOW_20=<enum>
,ECFLIST_STARTTIME_20=<string>
,ECFLIST_STOPTIME_20=<string>
,ECFLIST_CALLFWDTYPE_20=<enum>
,ECFLIST_SCREENOPTION_20=<enum>
,ECFLIST_SERVICEID_20=<string>
,ECFLIST_DOW_21=<enum>
,ECFLIST_STARTTIME_21=<string>
,ECFLIST_STOPTIME_21=<string>
,ECFLIST_CALLFWDTYPE_21=<enum>
,ECFLIST_SCREENOPTION_21=<enum>
,ECFLIST_SERVICEID_21=<string>
,ECFLIST_DOW_22=<enum>
,ECFLIST_STARTTIME_22=<string>
,ECFLIST_STOPTIME_22=<string>
,ECFLIST_CALLFWDTYPE_22=<enum>
,ECFLIST_SCREENOPTION_22=<enum>
,ECFLIST_SERVICEID_22=<string>
,ECFLIST_DOW_23=<enum>
,ECFLIST_STARTTIME_23=<string>
,ECFLIST_STOPTIME_23=<string>
,ECFLIST_CALLFWDTYPE_23=<enum>
,ECFLIST_SCREENOPTION_23=<enum>
,ECFLIST_SERVICEID_23=<string>
,ECFLIST_DOW_24=<enum>
,ECFLIST_STARTTIME_24=<string>
,ECFLIST_STOPTIME_24=<string>
,ECFLIST_CALLFWDTYPE_24=<enum>
,ECFLIST_SCREENOPTION_24=<enum>
,ECFLIST_SERVICEID_24=<string>
```

```
,ECFLIST_DOW_25=<enum>
,ECFLIST_STARTTIME_25=<string>
,ECFLIST_STOPTIME_25=<string>
,ECFLIST_CALLFWDTYPE_25=<enum>
,ECFLIST_SCREENOPTION_25=<enum>
,ECFLIST_SERVICEID_25=<string>
,ECFLIST_DOW_26=<enum>
,ECFLIST_STARTTIME_26=<string>
,ECFLIST_STOPTIME_26=<string>
,ECFLIST_CALLFWDTYPE_26=<enum>
,ECFLIST_SCREENOPTION_26=<enum>
,ECFLIST_SERVICEID_26=<string>
,ECFLIST_DOW_27=<enum>
,ECFLIST_STARTTIME_27=<string>
,ECFLIST_STOPTIME_27=<string>
,ECFLIST_CALLFWDTYPE_27=<enum>
,ECFLIST_SCREENOPTION_27=<enum>
,ECFLIST_SERVICEID_27=<string>
,ECFLIST_DOW_28=<enum>
,ECFLIST_STARTTIME_28=<string>
,ECFLIST_STOPTIME_28=<string>
,ECFLIST_CALLFWDTYPE_28=<enum>
,ECFLIST_SCREENOPTION_28=<enum>
,ECFLIST_SERVICEID_28=<string>
,ECFLIST_DOW_29=<enum>
,ECFLIST_STARTTIME_29=<string>
,ECFLIST_STOPTIME_29=<string>
,ECFLIST_CALLFWDTYPE_29=<enum>
,ECFLIST_SCREENOPTION_29=<enum>
,ECFLIST_SERVICEID_29=<string>
,ECFLIST_DOW_30=<enum>
,ECFLIST_STARTTIME_30=<string>
,ECFLIST_STOPTIME_30=<string>
,ECFLIST_CALLFWDTYPE_30=<enum>
,ECFLIST_SCREENOPTION_30=<enum>
,ECFLIST_SERVICEID_30=<string>
,ECFLIST_DOW_31=<enum>
,ECFLIST_STARTTIME_31=<string>
,ECFLIST_STOPTIME_31=<string>
,ECFLIST_CALLFWDTYPE_31=<enum>
,ECFLIST_SCREENOPTION_31=<enum>
,ECFLIST_SERVICEID_31=<string>
,ECFLIST_DOW_32=<enum>
,ECFLIST_STARTTIME_32=<string>
,ECFLIST_STOPTIME_32=<string>
,ECFLIST_CALLFWDTYPE_32=<enum>
,ECFLIST_SCREENOPTION_32=<enum>
,ECFLIST_SERVICEID_32=<string>
,ECFLIST_DOW_33=<enum>
,ECFLIST_STARTTIME_33=<string>
,ECFLIST_STOPTIME_33=<string>
,ECFLIST_CALLFWDTYPE_33=<enum>
,ECFLIST_SCREENOPTION_33=<enum>
,ECFLIST_SERVICEID_33=<string>
,ECFLIST_DOW_34=<enum>
,ECFLIST_STARTTIME_34=<string>
,ECFLIST_STOPTIME_34=<string>
,ECFLIST_CALLFWDTYPE_34=<enum>
,ECFLIST_SCREENOPTION_34=<enum>
,ECFLIST_SERVICEID_34=<string>
,ECFLIST_DOW_35=<enum>
,ECFLIST_STARTTIME_35=<string>
```

```
,ECFLIST_STOPTIME_35=<string>
,ECFLIST_CALLFWDTYPE_35=<enum>
,ECFLIST_SCREENOPTION_35=<enum>
,ECFLIST_SERVICEID_35=<string>
,ECFLIST_DOW_36=<enum>
,ECFLIST_STARTTIME_36=<string>
,ECFLIST_STOPTIME_36=<string>
,ECFLIST_CALLFWDTYPE_36=<enum>
,ECFLIST_SCREENOPTION_36=<enum>
,ECFLIST_SERVICEID_36=<string>
,ECFLIST_DOW_37=<enum>
,ECFLIST_STARTTIME_37=<string>
,ECFLIST_STOPTIME_37=<string>
,ECFLIST_CALLFWDTYPE_37=<enum>
,ECFLIST_SCREENOPTION_37=<enum>
,ECFLIST_SERVICEID_37=<string>
,ECFLIST_DOW_38=<enum>
,ECFLIST_STARTTIME_38=<string>
,ECFLIST_STOPTIME_38=<string>
,ECFLIST_CALLFWDTYPE_38=<enum>
,ECFLIST_SCREENOPTION_38=<enum>
,ECFLIST_SERVICEID_38=<string>
,ECFLIST_DOW_39=<enum>
,ECFLIST_STARTTIME_39=<string>
,ECFLIST_STOPTIME_39=<string>
,ECFLIST_CALLFWDTYPE_39=<enum>
,ECFLIST_SCREENOPTION_39=<enum>
,ECFLIST_SERVICEID_39=<string>
,ECFLIST_DOW_40=<enum>
,ECFLIST_STARTTIME_40=<string>
,ECFLIST_STOPTIME_40=<string>
,ECFLIST_CALLFWDTYPE_40=<enum>
,ECFLIST_SCREENOPTION_40=<enum>
,ECFLIST_SERVICEID_40=<string>
,ECFLIST_DOW_41=<enum>
,ECFLIST_STARTTIME_41=<string>
,ECFLIST_STOPTIME_41=<string>
,ECFLIST_CALLFWDTYPE_41=<enum>
,ECFLIST_SCREENOPTION_41=<enum>
,ECFLIST_SERVICEID_41=<string>
,ECFLIST_DOW_42=<enum>
,ECFLIST_STARTTIME_42=<string>
,ECFLIST_STOPTIME_42=<string>
,ECFLIST_CALLFWDTYPE_42=<enum>
,ECFLIST_SCREENOPTION_42=<enum>
,ECFLIST_SERVICEID_42=<string>
,ECFLIST_DOW_43=<enum>
,ECFLIST_STARTTIME_43=<string>
,ECFLIST_STOPTIME_43=<string>
,ECFLIST_CALLFWDTYPE_43=<enum>
,ECFLIST_SCREENOPTION_43=<enum>
,ECFLIST_SERVICEID_43=<string>
,ECFLIST_DOW_44=<enum>
,ECFLIST_STARTTIME_44=<string>
,ECFLIST_STOPTIME_44=<string>
,ECFLIST_CALLFWDTYPE_44=<enum>
,ECFLIST_SCREENOPTION_44=<enum>
,ECFLIST_SERVICEID_44=<string>
,ECFLIST_DOW_45=<enum>
,ECFLIST_STARTTIME_45=<string>
,ECFLIST_STOPTIME_45=<string>
,ECFLIST_CALLFWDTYPE_45=<enum>
```

```
,ECFLIST_SCREENOPTION_45=<enum>
,ECFLIST_SERVICEID_45=<string>
,ECFLIST_DOW_46=<enum>
,ECFLIST_STARTTIME_46=<string>
,ECFLIST_STOPTIME_46=<string>
,ECFLIST_CALLFWDTYPE_46=<enum>
,ECFLIST_SCREENOPTION_46=<enum>
,ECFLIST_SERVICEID_46=<string>
,ECFLIST_DOW_47=<enum>
,ECFLIST_STARTTIME_47=<string>
,ECFLIST_STOPTIME_47=<string>
,ECFLIST_CALLFWDTYPE_47=<enum>
,ECFLIST_SCREENOPTION_47=<enum>
,ECFLIST_SERVICEID_47=<string>
,ECFLIST_DOW_48=<enum>
,ECFLIST_STARTTIME_48=<string>
,ECFLIST_STOPTIME_48=<string>
,ECFLIST_CALLFWDTYPE_48=<enum>
,ECFLIST_SCREENOPTION_48=<enum>
,ECFLIST_SERVICEID_48=<string>
,ECFLIST_DOW_49=<enum>
,ECFLIST_STARTTIME_49=<string>
,ECFLIST_STOPTIME_49=<string>
,ECFLIST_CALLFWDTYPE_49=<enum>
,ECFLIST_SCREENOPTION_49=<enum>
,ECFLIST_SERVICEID_49=<string>
,NO_OF_RINGS=<int>
,RING_DURATION=<int>
</CFECF>

<CFEACR>
EACR -  (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
,SUBCONTROLLED=<bool>
,BILLING=<enum>
,EACRDEST=<enum>
,VOICEMAILDN=<string>
</CFEACR>

<CFEXO>
EXO - "Executive Busy Override" (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
,EXOOPTION=<enum>
</CFEXO>

<CFHotDesk>
HotDesk -  (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
,HOTDESKSIDE=<enum>
</CFHotDesk>

<CFLINR>
LINR - Last Incoming Number Redial (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
</CFLINR>

<CFLONR>
LONR - Last Outgoing Number Redial (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
```

```
</CFLONR>

<CFMCT>
MCT - "Customer Originated Trace" (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,BILLING=<enum>
</CFMCT>

<CFMOH>
MOH - "Music On Hold" (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
,INTERCEPTNAME=<string>
</CFMOH>

<CFONS>
ONS - "One Number Service" (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
,ONSTYPE=<enum>
,STATICONDACTIVATE=<bool>
,STATICONDSERVICEID=<string>
</CFONS>

<CFRACF>
RACF - "Remote Activation Call Forward" (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,PIN=<string>
</CFRACF>

<CFRCF>
RCF - "Remote Call forward" (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,SERVICEID=<string>
,MAXSIMULTANEOUSCALLS=<int>
</CFRCF>

<CFRFA>
RFA - Remote Feature Access (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,PIN=<string>
</CFRFA>

<CFSCA>
SCA - Selective Caller Accept (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
,BILLING=<enum>
,SCREENLIST_LEN=<int>
,SERVICEID_0=<string>
,SUBSCRIBERPPS_0=<enum>
,SERVICEID_1=<string>
,SUBSCRIBERPPS_1=<enum>
,SERVICEID_2=<string>
,SUBSCRIBERPPS_2=<enum>
,SERVICEID_3=<string>
,SUBSCRIBERPPS_3=<enum>
,SERVICEID_4=<string>
,SUBSCRIBERPPS_4=<enum>
,SERVICEID_5=<string>
,SUBSCRIBERPPS_5=<enum>
,SERVICEID_6=<string>
```

```
,SUBSCRIBERPPS_6=<enum>
,SERVICEID_7=<string>
,SUBSCRIBERPPS_7=<enum>
,SERVICEID_8=<string>
,SUBSCRIBERPPS_8=<enum>
,SERVICEID_9=<string>
,SUBSCRIBERPPS_9=<enum>
,SERVICEID_10=<string>
,SUBSCRIBERPPS_10=<enum>
,SERVICEID_11=<string>
,SUBSCRIBERPPS_11=<enum>
,SERVICEID_12=<string>
,SUBSCRIBERPPS_12=<enum>
,SERVICEID_13=<string>
,SUBSCRIBERPPS_13=<enum>
,SERVICEID_14=<string>
,SUBSCRIBERPPS_14=<enum>
,SERVICEID_15=<string>
,SUBSCRIBERPPS_15=<enum>
,SERVICEID_16=<string>
,SUBSCRIBERPPS_16=<enum>
,SERVICEID_17=<string>
,SUBSCRIBERPPS_17=<enum>
,SERVICEID_18=<string>
,SUBSCRIBERPPS_18=<enum>
,SERVICEID_19=<string>
,SUBSCRIBERPPS_19=<enum>
,SERVICEID_20=<string>
,SUBSCRIBERPPS_20=<enum>
,SERVICEID_21=<string>
,SUBSCRIBERPPS_21=<enum>
,SERVICEID_22=<string>
,SUBSCRIBERPPS_22=<enum>
,SERVICEID_23=<string>
,SUBSCRIBERPPS_23=<enum>
,SERVICEID_24=<string>
,SUBSCRIBERPPS_24=<enum>
,SERVICEID_25=<string>
,SUBSCRIBERPPS_25=<enum>
,SERVICEID_26=<string>
,SUBSCRIBERPPS_26=<enum>
,SERVICEID_27=<string>
,SUBSCRIBERPPS_27=<enum>
,SERVICEID_28=<string>
,SUBSCRIBERPPS_28=<enum>
,SERVICEID_29=<string>
,SUBSCRIBERPPS_29=<enum>
,SERVICEID_30=<string>
,SUBSCRIBERPPS_30=<enum>
,SERVICEID_31=<string>
,SUBSCRIBERPPS_31=<enum>
,PIN=<string>
,SCATERMINATIONTREATMENT=<enum>
,SERVICEID=<string>
</CFSCA>

<CFSCC>
SCC - "Station Controlled Conference service definition"
(Please see WSDL API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
</CFSCC>
```

```
<CFSCF>
SCF - Selective Call Forward (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
,SERVICEID=<string>
,SPLASHRING=<bool>
,BLACKLIST=<bool>
,SCREENLIST_LEN=<int>
,SERVICEID_0=<string>
,SUBSCRIBERPPS_0=<enum>
,SERVICEID_1=<string>
,SUBSCRIBERPPS_1=<enum>
,SERVICEID_2=<string>
,SUBSCRIBERPPS_2=<enum>
,SERVICEID_3=<string>
,SUBSCRIBERPPS_3=<enum>
,SERVICEID_4=<string>
,SUBSCRIBERPPS_4=<enum>
,SERVICEID_5=<string>
,SUBSCRIBERPPS_5=<enum>
,SERVICEID_6=<string>
,SUBSCRIBERPPS_6=<enum>
,SERVICEID_7=<string>
,SUBSCRIBERPPS_7=<enum>
,SERVICEID_8=<string>
,SUBSCRIBERPPS_8=<enum>
,SERVICEID_9=<string>
,SUBSCRIBERPPS_9=<enum>
,SERVICEID_10=<string>
,SUBSCRIBERPPS_10=<enum>
,SERVICEID_11=<string>
,SUBSCRIBERPPS_11=<enum>
,SERVICEID_12=<string>
,SUBSCRIBERPPS_12=<enum>
,SERVICEID_13=<string>
,SUBSCRIBERPPS_13=<enum>
,SERVICEID_14=<string>
,SUBSCRIBERPPS_14=<enum>
,SERVICEID_15=<string>
,SUBSCRIBERPPS_15=<enum>
,SERVICEID_16=<string>
,SUBSCRIBERPPS_16=<enum>
,SERVICEID_17=<string>
,SUBSCRIBERPPS_17=<enum>
,SERVICEID_18=<string>
,SUBSCRIBERPPS_18=<enum>
,SERVICEID_19=<string>
,SUBSCRIBERPPS_19=<enum>
,SERVICEID_20=<string>
,SUBSCRIBERPPS_20=<enum>
,SERVICEID_21=<string>
,SUBSCRIBERPPS_21=<enum>
,SERVICEID_22=<string>
,SUBSCRIBERPPS_22=<enum>
,SERVICEID_23=<string>
,SUBSCRIBERPPS_23=<enum>
,SERVICEID_24=<string>
,SUBSCRIBERPPS_24=<enum>
,SERVICEID_25=<string>
,SUBSCRIBERPPS_25=<enum>
,SERVICEID_26=<string>
,SUBSCRIBERPPS_26=<enum>
```

```
,SERVICEID_27=<string>
,SUBSCRIBERPPS_27=<enum>
,SERVICEID_28=<string>
,SUBSCRIBERPPS_28=<enum>
,SERVICEID_29=<string>
,SUBSCRIBERPPS_29=<enum>
,SERVICEID_30=<string>
,SUBSCRIBERPPS_30=<enum>
,SERVICEID_31=<string>
,SUBSCRIBERPPS_31=<enum>
</CFSCF>

<CFSCR>
SCR - Selective Caller Reject (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
,SUBCONTROLLED=<bool>
,BILLING=<enum>
,SCREENLIST_LEN=<int>
,SERVICEID_0=<string>
,SUBSCRIBERPPS_0=<enum>
,SERVICEID_1=<string>
,SUBSCRIBERPPS_1=<enum>
,SERVICEID_2=<string>
,SUBSCRIBERPPS_2=<enum>
,SERVICEID_3=<string>
,SUBSCRIBERPPS_3=<enum>
,SERVICEID_4=<string>
,SUBSCRIBERPPS_4=<enum>
,SERVICEID_5=<string>
,SUBSCRIBERPPS_5=<enum>
,SERVICEID_6=<string>
,SUBSCRIBERPPS_6=<enum>
,SERVICEID_7=<string>
,SUBSCRIBERPPS_7=<enum>
,SERVICEID_8=<string>
,SUBSCRIBERPPS_8=<enum>
,SERVICEID_9=<string>
,SUBSCRIBERPPS_9=<enum>
,SERVICEID_10=<string>
,SUBSCRIBERPPS_10=<enum>
,SERVICEID_11=<string>
,SUBSCRIBERPPS_11=<enum>
,SERVICEID_12=<string>
,SUBSCRIBERPPS_12=<enum>
,SERVICEID_13=<string>
,SUBSCRIBERPPS_13=<enum>
,SERVICEID_14=<string>
,SUBSCRIBERPPS_14=<enum>
,SERVICEID_15=<string>
,SUBSCRIBERPPS_15=<enum>
,SERVICEID_16=<string>
,SUBSCRIBERPPS_16=<enum>
,SERVICEID_17=<string>
,SUBSCRIBERPPS_17=<enum>
,SERVICEID_18=<string>
,SUBSCRIBERPPS_18=<enum>
,SERVICEID_19=<string>
,SUBSCRIBERPPS_19=<enum>
,SERVICEID_20=<string>
,SUBSCRIBERPPS_20=<enum>
,SERVICEID_21=<string>
```

```
,SUBSCRIBERPPS_21=<enum>
,SERVICEID_22=<string>
,SUBSCRIBERPPS_22=<enum>
,SERVICEID_23=<string>
,SUBSCRIBERPPS_23=<enum>
,SERVICEID_24=<string>
,SUBSCRIBERPPS_24=<enum>
,SERVICEID_25=<string>
,SUBSCRIBERPPS_25=<enum>
,SERVICEID_26=<string>
,SUBSCRIBERPPS_26=<enum>
,SERVICEID_27=<string>
,SUBSCRIBERPPS_27=<enum>
,SERVICEID_28=<string>
,SUBSCRIBERPPS_28=<enum>
,SERVICEID_29=<string>
,SUBSCRIBERPPS_29=<enum>
,SERVICEID_30=<string>
,SUBSCRIBERPPS_30=<enum>
,SERVICEID_31=<string>
,SUBSCRIBERPPS_31=<enum>
</CFSCR>

<CFSERRNG>
SERRNG - Serial Ringing (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
,SERIALDNLIST_LEN=<int>
,DURATION=<int>
,SERVICEID_0=<string>
,DURATION_0=<int>
,DNACTIVE_0=<bool>
,SUBSCRIBERPPS_0=<enum>
,EXTENSION_0=<bool>
,SERVICEID_1=<string>
,DURATION_1=<int>
,DNACTIVE_1=<bool>
,SUBSCRIBERPPS_1=<enum>
,EXTENSION_1=<bool>
,SERVICEID_2=<string>
,DURATION_2=<int>
,DNACTIVE_2=<bool>
,SUBSCRIBERPPS_2=<enum>
,EXTENSION_2=<bool>
,SERVICEID_3=<string>
,DURATION_3=<int>
,DNACTIVE_3=<bool>
,SUBSCRIBERPPS_3=<enum>
,EXTENSION_3=<bool>
,SERVICEID_4=<string>
,DURATION_4=<int>
,DNACTIVE_4=<bool>
,SUBSCRIBERPPS_4=<enum>
,EXTENSION_4=<bool>
,SERVICEID_5=<string>
,DURATION_5=<int>
,DNACTIVE_5=<bool>
,SUBSCRIBERPPS_5=<enum>
,EXTENSION_5=<bool>
</CFSERRNG>

<CFSILM>
```

```
   CFSILM
   ,ENABLED=<bool>
</CFSILM>

<CFSILMProtection>
   CFSILMProtection
   ,ENABLED=<bool>
</CFSILMProtection>

<CFSILMTag>
   CFSILMTag
   ,ENABLED=<bool>
</CFSILMTag>

<CFSPCALL>
SPCALL - "Speed call Feature" (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,ONEDIGIT=<string>
,TWODIGIT=<string>
,ONEDIGIT_LIST_LEN=<int>
,ONEDIGIT_SERVICEID_0=<string>
,ONEDIGIT_SPCALLNUM_0=<int>
,ONEDIGIT_SERVICEID_1=<string>
,ONEDIGIT_SPCALLNUM_1=<int>
,ONEDIGIT_SERVICEID_2=<string>
,ONEDIGIT_SPCALLNUM_2=<int>
,ONEDIGIT_SERVICEID_3=<string>
,ONEDIGIT_SPCALLNUM_3=<int>
,ONEDIGIT_SERVICEID_4=<string>
,ONEDIGIT_SPCALLNUM_4=<int>
,ONEDIGIT_SERVICEID_5=<string>
,ONEDIGIT_SPCALLNUM_5=<int>
,ONEDIGIT_SERVICEID_6=<string>
,ONEDIGIT_SPCALLNUM_6=<int>
,ONEDIGIT_SERVICEID_7=<string>
,ONEDIGIT_SPCALLNUM_7=<int>
,ONEDIGIT_SERVICEID_8=<string>
,ONEDIGIT_SPCALLNUM_8=<int>
,ONEDIGIT_SERVICEID_9=<string>
,ONEDIGIT_SPCALLNUM_9=<int>
,ONEDIGIT_SERVICEID_10=<string>
,ONEDIGIT_SPCALLNUM_10=<int>
,ONEDIGIT_SERVICEID_11=<string>
,ONEDIGIT_SPCALLNUM_11=<int>
,ONEDIGIT_SERVICEID_12=<string>
,ONEDIGIT_SPCALLNUM_12=<int>
,ONEDIGIT_SERVICEID_13=<string>
,ONEDIGIT_SPCALLNUM_13=<int>
,ONEDIGIT_SERVICEID_14=<string>
,ONEDIGIT_SPCALLNUM_14=<int>
,ONEDIGIT_SERVICEID_15=<string>
,ONEDIGIT_SPCALLNUM_15=<int>
,ONEDIGIT_SERVICEID_16=<string>
,ONEDIGIT_SPCALLNUM_16=<int>
,ONEDIGIT_SERVICEID_17=<string>
,ONEDIGIT_SPCALLNUM_17=<int>
,ONEDIGIT_SERVICEID_18=<string>
,ONEDIGIT_SPCALLNUM_18=<int>
,ONEDIGIT_SERVICEID_19=<string>
,ONEDIGIT_SPCALLNUM_19=<int>
,ONEDIGIT_SERVICEID_20=<string>
,ONEDIGIT_SPCALLNUM_20=<int>
```

```
,ONEDIGIT_SERVICEID_21=<string>
,ONEDIGIT_SPCALLNUM_21=<int>
,ONEDIGIT_SERVICEID_22=<string>
,ONEDIGIT_SPCALLNUM_22=<int>
,ONEDIGIT_SERVICEID_23=<string>
,ONEDIGIT_SPCALLNUM_23=<int>
,ONEDIGIT_SERVICEID_24=<string>
,ONEDIGIT_SPCALLNUM_24=<int>
,ONEDIGIT_SERVICEID_25=<string>
,ONEDIGIT_SPCALLNUM_25=<int>
,ONEDIGIT_SERVICEID_26=<string>
,ONEDIGIT_SPCALLNUM_26=<int>
,ONEDIGIT_SERVICEID_27=<string>
,ONEDIGIT_SPCALLNUM_27=<int>
,ONEDIGIT_SERVICEID_28=<string>
,ONEDIGIT_SPCALLNUM_28=<int>
,ONEDIGIT_SERVICEID_29=<string>
,ONEDIGIT_SPCALLNUM_29=<int>
,TWODIGIT_LIST_LEN=<int>
,TWODIGIT_SERVICEID_0=<string>
,TWODIGIT_SPCALLNUM_0=<int>
,TWODIGIT_SERVICEID_1=<string>
,TWODIGIT_SPCALLNUM_1=<int>
,TWODIGIT_SERVICEID_2=<string>
,TWODIGIT_SPCALLNUM_2=<int>
,TWODIGIT_SERVICEID_3=<string>
,TWODIGIT_SPCALLNUM_3=<int>
,TWODIGIT_SERVICEID_4=<string>
,TWODIGIT_SPCALLNUM_4=<int>
,TWODIGIT_SERVICEID_5=<string>
,TWODIGIT_SPCALLNUM_5=<int>
,TWODIGIT_SERVICEID_6=<string>
,TWODIGIT_SPCALLNUM_6=<int>
,TWODIGIT_SERVICEID_7=<string>
,TWODIGIT_SPCALLNUM_7=<int>
,TWODIGIT_SERVICEID_8=<string>
,TWODIGIT_SPCALLNUM_8=<int>
,TWODIGIT_SERVICEID_9=<string>
,TWODIGIT_SPCALLNUM_9=<int>
,TWODIGIT_SERVICEID_10=<string>
,TWODIGIT_SPCALLNUM_10=<int>
,TWODIGIT_SERVICEID_11=<string>
,TWODIGIT_SPCALLNUM_11=<int>
,TWODIGIT_SERVICEID_12=<string>
,TWODIGIT_SPCALLNUM_12=<int>
,TWODIGIT_SERVICEID_13=<string>
,TWODIGIT_SPCALLNUM_13=<int>
,TWODIGIT_SERVICEID_14=<string>
,TWODIGIT_SPCALLNUM_14=<int>
,TWODIGIT_SERVICEID_15=<string>
,TWODIGIT_SPCALLNUM_15=<int>
,TWODIGIT_SERVICEID_16=<string>
,TWODIGIT_SPCALLNUM_16=<int>
,TWODIGIT_SERVICEID_17=<string>
,TWODIGIT_SPCALLNUM_17=<int>
,TWODIGIT_SERVICEID_18=<string>
,TWODIGIT_SPCALLNUM_18=<int>
,TWODIGIT_SERVICEID_19=<string>
,TWODIGIT_SPCALLNUM_19=<int>
,TWODIGIT_SERVICEID_20=<string>
,TWODIGIT_SPCALLNUM_20=<int>
,TWODIGIT_SERVICEID_21=<string>
```

```
,TWODIGIT_SPCALLNUM_21=<int>
,TWODIGIT_SERVICEID_22=<string>
,TWODIGIT_SPCALLNUM_22=<int>
,TWODIGIT_SERVICEID_23=<string>
,TWODIGIT_SPCALLNUM_23=<int>
,TWODIGIT_SERVICEID_24=<string>
,TWODIGIT_SPCALLNUM_24=<int>
,TWODIGIT_SERVICEID_25=<string>
,TWODIGIT_SPCALLNUM_25=<int>
,TWODIGIT_SERVICEID_26=<string>
,TWODIGIT_SPCALLNUM_26=<int>
,TWODIGIT_SERVICEID_27=<string>
,TWODIGIT_SPCALLNUM_27=<int>
,TWODIGIT_SERVICEID_28=<string>
,TWODIGIT_SPCALLNUM_28=<int>
,TWODIGIT_SERVICEID_29=<string>
,TWODIGIT_SPCALLNUM_29=<int>
,SUBCONTROLLED=<bool>
</CFSPCALL>

<CFSR>
SR - Station Restriction     (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,ORIGINATINGSR=<enum>
,TERMINATINGSR=<enum>
</CFSR>

<CFSRS>
SRS -  (Please see WSDL API:unsp__UpdateSubscriberFeatures)
,DNRINGLIST_LEN=<int>
,ENABLED=<bool>
,SERVICEID_0=<string>
,EXTENSION_0=<bool>
,SERVICEID_1=<string>
,EXTENSION_1=<bool>
,SERVICEID_2=<string>
,EXTENSION_2=<bool>
,SERVICEID_3=<string>
,EXTENSION_3=<bool>
,SERVICEID_4=<string>
,EXTENSION_4=<bool>
,SERVICEID_5=<string>
,EXTENSION_5=<bool>
</CFSRS>

<CFTRS>
TRS - "Toll Restrictions" (Please see WSDL
API:unsp__UpdateSubscriberFeatures in Provisioning Doc for
Attribute details)
,CORNAME=<string>
,ALTERNATECORNAME=<string>
,SWITCHOVERDURATION=<int>
,ALTCORDAYSCHEDULELIST_LEN=<int>
,ALTCORDAYSCHEDULELIST_DOW_0=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_0=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_0=<string>
,ALTCORDAYSCHEDULELIST_DOW_1=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_1=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_1=<string>
,ALTCORDAYSCHEDULELIST_DOW_2=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_2=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_2=<string>
```

```
,ALTCORDAYSCHEDULELIST_DOW_3=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_3=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_3=<string>
,ALTCORDAYSCHEDULELIST_DOW_4=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_4=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_4=<string>
,ALTCORDAYSCHEDULELIST_DOW_5=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_5=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_5=<string>
,ALTCORDAYSCHEDULELIST_DOW_6=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_6=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_6=<string>
,ALTCORDAYSCHEDULELIST_DOW_7=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_7=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_7=<string>
,ALTCORDAYSCHEDULELIST_DOW_8=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_8=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_8=<string>
,ALTCORDAYSCHEDULELIST_DOW_9=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_9=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_9=<string>
,ALTCORDAYSCHEDULELIST_DOW_10=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_10=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_10=<string>
,ALTCORDAYSCHEDULELIST_DOW_11=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_11=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_11=<string>
,ALTCORDAYSCHEDULELIST_DOW_12=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_12=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_12=<string>
,ALTCORDAYSCHEDULELIST_DOW_13=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_13=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_13=<string>
,ALTCORDAYSCHEDULELIST_DOW_14=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_14=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_14=<string>
,ALTCORDAYSCHEDULELIST_DOW_15=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_15=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_15=<string>
,ALTCORDAYSCHEDULELIST_DOW_16=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_16=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_16=<string>
,ALTCORDAYSCHEDULELIST_DOW_17=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_17=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_17=<string>
,ALTCORDAYSCHEDULELIST_DOW_18=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_18=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_18=<string>
,ALTCORDAYSCHEDULELIST_DOW_19=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_19=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_19=<string>
,ALTCORDAYSCHEDULELIST_DOW_20=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_20=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_20=<string>
,ALTCORDAYSCHEDULELIST_DOW_21=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_21=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_21=<string>
,ALTCORDAYSCHEDULELIST_DOW_22=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_22=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_22=<string>
,ALTCORDAYSCHEDULELIST_DOW_23=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_23=<string>
```

```
,ALTCORDAYSCHEDULELIST_STOPTIME_23=<string>
,ALTCORDAYSCHEDULELIST_DOW_24=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_24=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_24=<string>
,ALTCORDAYSCHEDULELIST_DOW_25=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_25=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_25=<string>
,ALTCORDAYSCHEDULELIST_DOW_26=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_26=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_26=<string>
,ALTCORDAYSCHEDULELIST_DOW_27=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_27=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_27=<string>
,ALTCORDAYSCHEDULELIST_DOW_28=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_28=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_28=<string>
,ALTCORDAYSCHEDULELIST_DOW_29=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_29=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_29=<string>
,ALTCORDAYSCHEDULELIST_DOW_30=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_30=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_30=<string>
,ALTCORDAYSCHEDULELIST_DOW_31=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_31=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_31=<string>
,ALTCORDAYSCHEDULELIST_DOW_32=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_32=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_32=<string>
,ALTCORDAYSCHEDULELIST_DOW_33=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_33=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_33=<string>
,ALTCORDAYSCHEDULELIST_DOW_34=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_34=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_34=<string>
,ALTCORDAYSCHEDULELIST_DOW_35=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_35=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_35=<string>
,ALTCORDAYSCHEDULELIST_DOW_36=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_36=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_36=<string>
,ALTCORDAYSCHEDULELIST_DOW_37=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_37=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_37=<string>
,ALTCORDAYSCHEDULELIST_DOW_38=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_38=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_38=<string>
,ALTCORDAYSCHEDULELIST_DOW_39=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_39=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_39=<string>
,ALTCORDAYSCHEDULELIST_DOW_40=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_40=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_40=<string>
,ALTCORDAYSCHEDULELIST_DOW_41=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_41=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_41=<string>
,ALTCORDAYSCHEDULELIST_DOW_42=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_42=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_42=<string>
,ALTCORDAYSCHEDULELIST_DOW_43=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_43=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_43=<string>
,ALTCORDAYSCHEDULELIST_DOW_44=<enum>
```

```
,ALTCORDAYSCHEDULELIST_STARTTIME_44=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_44=<string>
,ALTCORDAYSCHEDULELIST_DOW_45=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_45=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_45=<string>
,ALTCORDAYSCHEDULELIST_DOW_46=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_46=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_46=<string>
,ALTCORDAYSCHEDULELIST_DOW_47=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_47=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_47=<string>
,ALTCORDAYSCHEDULELIST_DOW_48=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_48=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_48=<string>
,ALTCORDATESCHEDULELIST_LEN=<int>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_0=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_0=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_1=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_1=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_2=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_2=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_3=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_3=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_4=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_4=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_5=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_5=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_6=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_6=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_7=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_7=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_8=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_8=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_9=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_9=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_10=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_10=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_11=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_11=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_12=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_12=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_13=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_13=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_14=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_14=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_15=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_15=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_16=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_16=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_17=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_17=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_18=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_18=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_19=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_19=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_20=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_20=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_21=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_21=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_22=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_22=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_23=<string>
```

```
,ALTCORDATESCHEDULELIST_ENDTIME_23=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_24=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_24=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_25=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_25=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_26=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_26=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_27=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_27=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_28=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_28=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_29=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_29=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_30=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_30=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_31=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_31=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_32=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_32=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_33=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_33=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_34=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_34=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_35=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_35=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_36=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_36=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_37=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_37=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_38=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_38=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_39=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_39=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_40=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_40=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_41=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_41=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_42=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_42=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_43=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_43=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_44=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_44=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_45=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_45=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_46=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_46=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_47=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_47=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_48=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_48=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_49=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_49=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_50=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_50=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_51=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_51=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_52=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_52=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_53=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_53=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_54=<string>
```

```
,ALTCORDATESCHEDULELIST_ENDTIME_54=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_55=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_55=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_56=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_56=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_57=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_57=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_58=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_58=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_59=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_59=<string>
,RESTRICTEDLINELIST_LEN=<int>
,SERVICEID_0=<string>
,SERVICEID_1=<string>
,SERVICEID_2=<string>
,SERVICEID_3=<string>
,SERVICEID_4=<string>
,SERVICEID_5=<string>
,SERVICEID_6=<string>
,SERVICEID_7=<string>
,SERVICEID_8=<string>
,SERVICEID_9=<string>
</CFTRS>

<CFVM>
VM - Voice Mail (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
,SERVICEID=<string>
,NUMBEROFRINGS=<int>
,VMWI_=<bool>
,AMWI_=<bool>
,RINGDURATION=<int>
</CFVM>

<CFWL>
WL -  (Please see WSDL API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
,WARMLINEDN=<string>
,SUBACTIVATABLE=<enum>
</CFWL>
```

24.2.2 Remove Features

```
<RF>
RF - Remove Features from subscribers (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
   ,SUBSCRIBERDN=<string>
   feat1
   feat2
   .
   .
   <featn>
</RF>
```

24.2.3 Modify Features

```
<MF>
MF - Modify Features for subscribers (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
   ,SUBSCRIBERDN=<string>
   feat1
,feat1_attribute1=<value>
,feat1_attribute2=<value>
   feat2
,feat2_attribute1=<value>
,feat2_attribute2=<value>
   .
   .
   featn
,featn_attribute1=<value>
,featn_attribute2=<value>
</MF>
```

24.2.4 Deny Features

```
<DF>
DF - Deny Features for subscribers (Please see WSDL
API:unsp__UpdateSubscriberFeatures)
   ,SUBSCRIBERDN=<string>
   feat1
   feat2
   .
   .
   <featn>
</DF>
```

24.2.5 Add Feature Profile Features

```
<FPAF>
FPAF - Add Features to Feature Profile
   ,FPNAME=<string>
   ,BGNAME=<string>
   feat1
,feat1_attribute1=<value>
,feat1_attribute2=<value>
   feat2
,feat2_attribute1=<value>
,feat2_attribute2=<value>
   .
   .
   featn
,featn_attribute1=<value>
,featn_attribute2=<value>
</FPAF>
```

24.2.6 Remove Feature Profile Features

```
<FPRF>
FPRF - Remove Features from Feature Profile
   ,FPNAME=<string>
   ,BGNAME=<string>
   feat1
   feat2
   .
   .
   featn
</FPRF>
```

24.2.7 Modify Feature Profile Features

```
<FPMF>
FPMF - Modify Features in a Feature Profile
   ,FPNAME=<string>
   ,BGNAME=<string>
   feat1
,feat1_attribute1=<value>
,feat1_attribute2=<value>
   feat2
,feat2_attribute1=<value>
,feat2_attribute2=<value>
   .
   .
   featn
,featn_attribute1=<value>
,featn_attribute2=<value>
</FPMF>
```

24.2.8 Add Business Group Features

```
<ABF>
ABF - Add Business Group Features
   ,BGNAME=<string>
   feat1
,feat1_attribute1=<value>
,feat1_attribute2=<value>
   feat2
,feat2_attribute1=<value>
,feat2_attribute2=<value>
   .
   .
   featn
,featn_attribute1=<value>
,featn_attribute2=<value>
</ABF>
```

24.2.8.1 Supported Features

Following is a list of features supported by the ABF command above along with their attributes. Note that these features are also supported by the RBF (Remove Business Group Features) command.

```
<CFBgDR>
BgDR - "Intra BG Distinctive Ringing" (Please see WSDL
API:unsp__UpdateBGFeatures)
,ENABLED=<bool>
</CFBgDR>

<CFBgE911LIN>
BgE911LIN - "BG E911 LIN Administration" (Please see WSDL
API:unsp__UpdateBGFeatures)
,DEFAULTEMERGENCYNUMBER=<string>
,DEFAULTINTERCEPTNAME=<string>
</CFBgE911LIN>

<CFBgHotDesk>
BgHotDesk - "The Hot Desking Configuration Service" (Please
see WSDL API:unsp__UpdateBGFeatures)
,AUTOLOGOFFTIME=<string>
</CFBgHotDesk>

<CFBgKS>
BgKS - "Keyset Operation BG only" (Please see WSDL
API:unsp__UpdateBGFeatures)
,MANUALHOLDRECALLTIMER=<int>
,LINERESERVATIONTIMER=<int>
</CFBgKS>

<CFBgLC>
BgLC - "Business Group Large Conference service definition"
(Please see WSDL API:unsp__UpdateBGFeatures)
,MINNUMBERPORTS=<int>
,MAXNUMBERPORTS=<int>
,MAXCONFERENCES=<int>
,MAXCONFERENCEMEMBERS=<int>
,CONFERENCERECALLDISCTIMER=<int>
,PLAYENTRYEXITTONE=<bool>
,BLOCKLCONFRESTRICTEDPARTY=<bool>
</CFBgLC>

<CFBgNBCPU>
BgNBCPU - "Business Group Night Bell CPU service definition"
(Please see WSDL API:unsp__UpdateBGFeatures)
,ENABLED=<bool>
,NIGHTBELLCPUID=<int>
</CFBgNBCPU>

<CFBgSILM >
BgSILM  - "The Hot Desking Configuration Service" (Please
see WSDL API:unsp__UpdateBGFeatures)
, MONITORTAGLIMIT <int>
, MAXNUMBEROFSUPERVISORS <int>
</CFBgSILM >

<CFBgSpDial>
```

```
CFBgSpDial - "Bg System Speed Calling Lists" (Please see
WSDL API:unsp__UpdateSubscriberFeatures)
,ENABLED=<bool>
,LISTIDONE=<int>
,LISTIDTWO=<int>
</CFBgSpDial>

<CFBGTM>
    CFBGTM
</CFBGTM>
```

24.2.9 Remove Business Group Features

```
<RBF>
RBF - Remove Business Group Features
   ,BGNAME=<string>
   feat1
   feat2
   .
   .
   featn
</RBF>
```

24.2.10 Add End Point Profile Features

```
<EPADDFEAT>
EPADDFEAT - Add End Point Profile Features
   ,EPNAME=<string>
   feat1
,feat1_attribute1=<value>
,feat1_attribute2=<value>
   feat2
,feat2_attribute1=<value>
,feat2_attribute2=<value>
   .
   .
   featn
,featn_attribute1=<value>
,featn_attribute2=<value>
</EPADDFEAT>
```

24.2.10.1 Supported Features

Following is a list of features supported by the EPADDFEAT command above along with their attributes. Note that these features are also supported by the DPDELFEAT (Remove Endpoint Profile Features) command.

```
<CFCFID>
CFID - "Call Forward Invalid Destination" (Please see WSDL
```

```
API:unsp__UpdateEpFeatures in Provisioning Doc for
Attribute details)
,ENABLED=<bool>
,ALTERNATETARGET=<string>
</CFCFID>

<CFEpCNAM>
EpCNAM - "Incoming Calling Name Delivery" (Please see WSDL
API:unsp__UpdateEpFeatures)
</CFEpCNAM>

<CFEPCISNAME>
EPCISNAME - Called Party Name Delivery (Please see WSDL
API:unsp__UpdateEpFeatures)
,NAMEDB=<enum>
</CFEPCISNAME>

<CFEpCT>
EpCT - "Endpoint Call Transfer" (Please see WSDL
API:unsp__UpdateEpFeatures in Provisioning Doc for
Attribute details)
,ENABLED=<bool>
</CFEpCT>

<CFEPTRS>
EPTRS - "Toll Restrictions" (Please see WSDL
API:unsp__UpdateEpFeatures in Provisioning Doc for
Attribute details)
,CORNAME=<string>
,ALTERNATECORNAME=<string>
,SWITCHOVERDURATION=<int>
,ALTCORDAYSCHEDULELIST_LEN=<int>
,ALTCORDAYSCHEDULELIST_DOW_0=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_0=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_0=<string>
,ALTCORDAYSCHEDULELIST_DOW_1=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_1=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_1=<string>
,ALTCORDAYSCHEDULELIST_DOW_2=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_2=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_2=<string>
,ALTCORDAYSCHEDULELIST_DOW_3=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_3=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_3=<string>
,ALTCORDAYSCHEDULELIST_DOW_4=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_4=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_4=<string>
,ALTCORDAYSCHEDULELIST_DOW_5=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_5=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_5=<string>
,ALTCORDAYSCHEDULELIST_DOW_6=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_6=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_6=<string>
,ALTCORDAYSCHEDULELIST_DOW_7=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_7=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_7=<string>
,ALTCORDAYSCHEDULELIST_DOW_8=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_8=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_8=<string>
,ALTCORDAYSCHEDULELIST_DOW_9=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_9=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_9=<string>
```

```
,ALTCORDAYSCHEDULELIST_DOW_10=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_10=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_10=<string>
,ALTCORDAYSCHEDULELIST_DOW_11=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_11=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_11=<string>
,ALTCORDAYSCHEDULELIST_DOW_12=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_12=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_12=<string>
,ALTCORDAYSCHEDULELIST_DOW_13=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_13=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_13=<string>
,ALTCORDAYSCHEDULELIST_DOW_14=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_14=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_14=<string>
,ALTCORDAYSCHEDULELIST_DOW_15=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_15=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_15=<string>
,ALTCORDAYSCHEDULELIST_DOW_16=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_16=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_16=<string>
,ALTCORDAYSCHEDULELIST_DOW_17=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_17=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_17=<string>
,ALTCORDAYSCHEDULELIST_DOW_18=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_18=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_18=<string>
,ALTCORDAYSCHEDULELIST_DOW_19=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_19=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_19=<string>
,ALTCORDAYSCHEDULELIST_DOW_20=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_20=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_20=<string>
,ALTCORDAYSCHEDULELIST_DOW_21=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_21=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_21=<string>
,ALTCORDAYSCHEDULELIST_DOW_22=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_22=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_22=<string>
,ALTCORDAYSCHEDULELIST_DOW_23=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_23=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_23=<string>
,ALTCORDAYSCHEDULELIST_DOW_24=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_24=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_24=<string>
,ALTCORDAYSCHEDULELIST_DOW_25=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_25=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_25=<string>
,ALTCORDAYSCHEDULELIST_DOW_26=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_26=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_26=<string>
,ALTCORDAYSCHEDULELIST_DOW_27=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_27=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_27=<string>
,ALTCORDAYSCHEDULELIST_DOW_28=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_28=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_28=<string>
,ALTCORDAYSCHEDULELIST_DOW_29=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_29=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_29=<string>
,ALTCORDAYSCHEDULELIST_DOW_30=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_30=<string>
```

```
,ALTCORDAYSCHEDULELIST_STOPTIME_30=<string>
,ALTCORDAYSCHEDULELIST_DOW_31=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_31=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_31=<string>
,ALTCORDAYSCHEDULELIST_DOW_32=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_32=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_32=<string>
,ALTCORDAYSCHEDULELIST_DOW_33=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_33=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_33=<string>
,ALTCORDAYSCHEDULELIST_DOW_34=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_34=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_34=<string>
,ALTCORDAYSCHEDULELIST_DOW_35=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_35=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_35=<string>
,ALTCORDAYSCHEDULELIST_DOW_36=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_36=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_36=<string>
,ALTCORDAYSCHEDULELIST_DOW_37=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_37=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_37=<string>
,ALTCORDAYSCHEDULELIST_DOW_38=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_38=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_38=<string>
,ALTCORDAYSCHEDULELIST_DOW_39=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_39=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_39=<string>
,ALTCORDAYSCHEDULELIST_DOW_40=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_40=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_40=<string>
,ALTCORDAYSCHEDULELIST_DOW_41=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_41=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_41=<string>
,ALTCORDAYSCHEDULELIST_DOW_42=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_42=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_42=<string>
,ALTCORDAYSCHEDULELIST_DOW_43=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_43=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_43=<string>
,ALTCORDAYSCHEDULELIST_DOW_44=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_44=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_44=<string>
,ALTCORDAYSCHEDULELIST_DOW_45=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_45=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_45=<string>
,ALTCORDAYSCHEDULELIST_DOW_46=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_46=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_46=<string>
,ALTCORDAYSCHEDULELIST_DOW_47=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_47=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_47=<string>
,ALTCORDAYSCHEDULELIST_DOW_48=<enum>
,ALTCORDAYSCHEDULELIST_STARTTIME_48=<string>
,ALTCORDAYSCHEDULELIST_STOPTIME_48=<string>
,ALTCORDATESCHEDULELIST_LEN=<int>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_0=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_0=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_1=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_1=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_2=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_2=<string>
```

```
,ALTCORDATESCHEDULELIST_STARTDATENTIME_3=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_3=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_4=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_4=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_5=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_5=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_6=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_6=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_7=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_7=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_8=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_8=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_9=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_9=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_10=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_10=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_11=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_11=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_12=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_12=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_13=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_13=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_14=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_14=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_15=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_15=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_16=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_16=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_17=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_17=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_18=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_18=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_19=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_19=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_20=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_20=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_21=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_21=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_22=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_22=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_23=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_23=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_24=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_24=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_25=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_25=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_26=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_26=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_27=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_27=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_28=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_28=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_29=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_29=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_30=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_30=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_31=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_31=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_32=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_32=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_33=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_33=<string>
```

```
,ALTCORDATESCHEDULELIST_STARTDATENTIME_34=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_34=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_35=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_35=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_36=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_36=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_37=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_37=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_38=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_38=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_39=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_39=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_40=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_40=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_41=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_41=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_42=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_42=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_43=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_43=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_44=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_44=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_45=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_45=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_46=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_46=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_47=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_47=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_48=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_48=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_49=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_49=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_50=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_50=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_51=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_51=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_52=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_52=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_53=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_53=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_54=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_54=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_55=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_55=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_56=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_56=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_57=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_57=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_58=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_58=<string>
,ALTCORDATESCHEDULELIST_STARTDATENTIME_59=<string>
,ALTCORDATESCHEDULELIST_ENDTIME_59=<string>
,RESTRICTEDLINELIST_LEN=<int>
,SERVICEID_0=<string>
,SERVICEID_1=<string>
,SERVICEID_2=<string>
,SERVICEID_3=<string>
,SERVICEID_4=<string>
,SERVICEID_5=<string>
,SERVICEID_6=<string>
,SERVICEID_7=<string>
,SERVICEID_8=<string>
```

```
,SERVICEID_9=<string>
</CFEPTRS

<CFSBRK>
SBRK - "Service Broker" (Please see WSDL
API:unsp__UpdateEpFeatures)
,ENABLED=<bool>
</CFSBRK>

<CFEPVM>
    CFEPVM
</CFEPVM>

<CFCNIM>
CNIM - "CSTA Network Interface Device" (Please see WSDL
API:unsp  UpdateEpFeatures in Provisioning Doc for
Attribute details)
,NAMEPROVIDER=<string>
</CFCNIM>
```

24.2.11 Create CSTA Application

```
<CCSTAAPP>
CCSTAAPP – "Creates a CSTA Application" (Please see WSDL
API:unsp__CreateCSTAApplication in Provisioning Doc for
Attribute details)
   ,APPNAME=<string>
   ,APPIP=<string>
   ,APPTYPE=<enum>,
   ,SOAPLISTENERPORT=<int>
   ,HEARTBEATTIMER=<int>
   ,HEARTBEATWATCHDOG=<int>
   ,HEARTBEATRETRYCOUNT=<int>
   ,APPID=<string>
   ,ENTERPRISEID=<string>
   ,PAS_CODE=<string>
   ,OUTSIDEACCESS=<int>
   ,BGNAME=<string>
   ,NPNAME=<string>
</CCSTAAPP>
```

24.2.12 Modify CSTA Application

```
<MCSTAAPP>
MCSTAAPP – "Modifies a CSTA Application" (Please see WSDL
API:unsp__ModifyCSTAApplication in Provisioning Doc for
Attribute details)
   ,APPIP=<string>
   ,APPNAME=<string>
   ,APPNEWIP=<string>
   ,APPTYPE=<enum>,
   ,SOAPLISTENERPORT=<int>
   ,HEARTBEATTIMER=<int>
   ,HEARTBEATWATCHDOG=<int>
   ,HEARTBEATRETRYCOUNT=<int>
```

```
   ,APPID=<string>
   ,ENTERPRISEID=<string>
   ,PAS_CODE=<string>
   ,OUTSIDEACCESS=<int>
   ,BGNAME=<string>
   ,NPNAME=<string>
</MCSTAAPP>
```

24.2.13 Delete CSTA Application

```
<DCSTAAPP>
DCSTAAPP - "Deletes a CSTA Application" (Please see WSDL
API:unsp__DeleteCSTAApplication in Provisioning Doc for
Attribute details)
   ,APPIP=<string>
</DCSTAAPP>
```

24.2.14 Delete End Point Profile Features

```
<EPDELFEAT>
EPDELFEAT - Delete End Point Profile Features
   ,EPNAME=<string>
   feat1
   feat2
   .
   .
   featn
</EPDELFEAT>
```

24.2.15 Create Subscriber

There a number of different kinds of subscribers that can be created. The commands for all of them are listed below.

```
<CS>
CS - Base Subscriber Profile.All parameters available for
all types of subscribers. (Please see WSDL
API:unsp__CreateSubscriber2)
,SUBSCRIBERDN=<string>
,RETAILER=<string>
,ACCOUNTID=<string>
,BILLINGID=<string>
,CLASSOFSERVICE=<string>
,RATEAREA=<string>
,CALLINGLOC=<string>
,TESTLINEFLAG=<bool>
,NUMBERPLANNAME=<string>
,BGNAME=<string>
,DISPLAYNAME=<string>
,UNICODEDISPLAYNAME=<string>
```

```
,TIMEZONE=<string>
,PIN=<string>
,PUBLICPIN=<string>
,SUBSCRIBERSTATUS=<enum>
,INTERCEPTANNOUNCEMENT=<enum>
,NEWSERVICEID=<string>
,LANGUAGE=<string>
,BGDEPTNAME=<string>
,BEARER_SPEECH=<bool>
,BEARER_3_1K=<bool>
,BEARER_56K=<bool>
,BEARER_64K=<bool>
,FAXMODE=<enum>
,ECHOCANCELLATION=<bool>
,SILENCESUPPRESSION=<bool>
,PIC_1=<string>
,PIC_2=<string>
,PIC_3=<string>
,PIC_4=<string>
,FAXSTATISTICS=<bool>
,OUTBANDDTMFSIGNALING=<bool>
,SUPPRESSOUTBANDDTMFWITHG711=<bool>
,QOSCLASSMODE=<enum>
,QOSVALUE=<int>
,QOSPROFILENAME=<string>
,QOSPROFILEBGNAME=<string>
,CODECPRIM=<enum>
,PPPRIM=<int>
,TOSPRIM=<int>
,CODECSEC=<enum>
,PPSEC=<int>
,TOSSEC=<int>
,CODECTHIRD=<enum>
,PPTHIRD=<int>
,TOSTHIRD=<int>
,CODECFOURTH=<enum>
,PPFOURTH=<int>
,TOSFOURTH=<int>
,EXTERNALDNFLAG=<bool>
,COUNTRYCODELEN=<int>
,FEATUREPROFILENAME=<string>
,FEATUREPROFILEBGNAME=<string>
,KEYSETUSE=<enum>
</CS>

<CS_SIP>
CS_SIP - Creates a SIP subscriber (Please see WSDL
API:unsp__CreateSubscriber2)
,SUBSCRIBERDN=<string>
,RETAILER=<string>
,ACCOUNTID=<string>
,BILLINGID=<string>
,CLASSOFSERVICE=<string>
,RATEAREA=<string>
,CALLINGLOC=<string>
,TESTLINEFLAG=<bool>
,NUMBERPLANNAME=<string>
,BGNAME=<string>
,DISPLAYNAME=<string>
,UNICODEDISPLAYNAME=<string>
,TIMEZONE=<string>
,PIN=<string>
```

```
,PIN2=<string>
,PIN3=<string>
,PIN4=<string>
,PIN5=<string>
,PUBLICPIN=<string>
,SUBSCRIBERSTATUS=<enum>
,INTERCEPTANNOUNCEMENT=<enum>
,NEWSERVICEID=<string>
,LANGUAGE=<string>
,BGDEPTNAME=<string>
,BEARER_SPEECH=<bool>
,BEARER_3_1K=<bool>
,BEARER_56K=<bool>
,BEARER_64K=<bool>
,FAXMODE=<enum>
,ECHOCANCELLATION=<bool>
,SILENCESUPPRESSION=<bool>
,PIC_1=<string>
,PIC_2=<string>
,PIC_3=<string>
,PIC_4=<string>
,FAXSTATISTICS=<bool>
,OUTBANDDTMFSIGNALING=<bool>
,SUPPRESSOUTBANDDTMFWITHG711=<bool>
,QOSPROFILENAME=<string>^M
,QOSPROFILEBGNAME=<string>^M
,EXTERNALDNFLAG=<bool>
,COUNTRYCODELEN=<int>
,FEATUREPROFILENAME=<string>
,FEATUREPROFILEBGNAME=<string>
,PHONENAME=<string>
,REGTYPE=<enum>
,SIGIPADDR=<string>
,SIGPORT=<int>
,LATADIGITS=<string>
,PREREGISTER=<bool>
,IPSEC_REALM=<string>
,IPSEC_UNAME=<string>
,IPSEC_PW=<string>
,IPSEC_SCHEME=<enum>
,IPSEC_SRTP=<enum>
,MAXSIPSESSIONS=<int>
,TRANSPORTTYPE=<enum>
,TRUSTEDSUB=<bool>
,GSMSUB=<bool>
,KEYSETUSE=<enum>
,ASSOCIATEDSIPENDPOINTNAME=<string>
,STATICIMSSUB=<bool>
,OVERRIDEIRMCODECRESTRICTIONS=<bool>
,LOOSEROUTINGURI=<string>
,MEDIAREDIRECTALLOWED=<bool>
,AORPERCONTACT=<bool>
,ALIASNAME0=<string>
,ALIASNAME1=<string>
,VIPTYPE=<enum>,
,SENDURITELEPHONESUBFORMAT=<bool>,
,SENDINTERNATIONALNUMBERGNF=<bool>
,RESERVED_2=<bool>
,RESERVED_3=<bool>
,RESERVED_4=<bool>
,DISLONGCALLAUDIT=<bool>
,ANATSUPPORT=<enum>,
```

```
,MOBILEFLAG=<bool>
,MAINDEVICEDN=<string>
,MOBRINGDUR=<int>
,MAINRINGDUR=<int>
,CELLRINGDUR=<int>

</CS_SIP>

<CS_MGCP>
CS_MGCP - Create MGCP subscriber
 (Please see WSDL API:unsp__CreateSubscriber2)
,SUBSCRIBERDN=<string>
,RETAILER=<string>
,ACCOUNTID=<string>
,BILLINGID=<string>
,CLASSOFSERVICE=<string>
,RATEAREA=<string>
,CALLINGLOC=<string>
,TESTLINEFLAG=<bool>
,NUMBERPLANNAME=<string>
,BGNAME=<string>
,DISPLAYNAME=<string>
,UNICODEDISPLAYNAME=<string>
,TIMEZONE=<string>
,PIN=<string>
,PUBLICPIN=<string>
,SUBSCRIBERSTATUS=<enum>
,INTERCEPTANNOUNCEMENT=<enum>
,NEWSERVICEID=<string>
,LANGUAGE=<string>
,BGDEPTNAME=<string>
,BEARER_SPEECH=<bool>
,BEARER_3_1K=<bool>
,BEARER_56K=<bool>
,BEARER_64K=<bool>
,FAXMODE=<enum>
,ECHOCANCELLATION=<bool>
,SILENCESUPPRESSION=<bool>
,PIC_1=<string>
,PIC_2=<string>
,PIC_3=<string>
,PIC_4=<string>
,FAXSTATISTICS=<bool>
,OUTBANDDTMFSIGNALING=<bool>
,SUPPRESSOUTBANDDTMFWITHG711=<bool>
,QOSCLASSMODE=<enum>
,QOSVALUE=<int>
,QOSPROFILENAME=<string>
,QOSPROFILEBGNAME=<string>
,CODECPRIM=<enum>
,PPPRIM=<int>
,TOSPRIM=<int>
,CODECSEC=<enum>
,PPSEC=<int>
,TOSSEC=<int>
,CODECTHIRD=<enum>
,PPTHIRD=<int>
,TOSTHIRD=<int>
,CODECFOURTH=<enum>
,PPFOURTH=<int>
,TOSFOURTH=<int>
,EXTERNALDNFLAG=<bool>
```

```
,COUNTRYCODELEN=<int>
,FEATUREPROFILENAME=<string>
,FEATUREPROFILEBGNAME=<string>
,DQTBRP=<int>
,DQTBSP=<int>
,DQPDRP=<int>
,DQMPUP=<int>
,DQMPSP=<int>
,DQRateP=<int>
,DQSTP=<int>
,DQTBRS=<int>
,DQTBSS=<int>
,DQPDRS=<int>
,DQMPUS=<int>
,DQMPSS=<int>
,DQRateS=<int>
,DQSTS=<int>
,DQTBRT=<int>
,DQTBST=<int>
,DQPDRT=<int>
,DQMPUT=<int>
,DQMPST=<int>
,DQRateT=<int>
,DQSTT=<int>
,DQTBRF=<int>
,DQTBSF=<int>
,DQPDRF=<int>
,DQMPUF=<int>
,DQMPSF=<int>
,DQRateF=<int>
,DQSTF=<int>
,MEDIAGATEWAYNAME=<string>
,MEDIAGATEWAYCIRCUIT=<string>
,LATADIGITS=<string>
,RTPSECURITY=<enum>
,KEYSETUSE=<enum>
</CS_MGCP>

<CS_H323>
CS_H323 - Create a H233 subscriber
 (Please see WSDL API:unsp__CreateSubscriber2)
,SUBSCRIBERDN=<string>
,RETAILER=<string>
,ACCOUNTID=<string>
,BILLINGID=<string>
,CLASSOFSERVICE=<string>
,RATEAREA=<string>
,CALLINGLOC=<string>
,TESTLINEFLAG=<bool>
,NUMBERPLANNAME=<string>
,BGNAME=<string>
,DISPLAYNAME=<string>
,UNICODEDISPLAYNAME=<string>
,TIMEZONE=<string>
,PIN=<string>
,PUBLICPIN=<string>
,SUBSCRIBERSTATUS=<enum>
,INTERCEPTANNOUNCEMENT=<enum>
,NEWSERVICEID=<string>
,LANGUAGE=<string>
,BGDEPTNAME=<string>
,BEARER_SPEECH=<bool>
```

```
,BEARER_3_1K=<bool>
,BEARER_56K=<bool>
,BEARER_64K=<bool>
,FAXMODE=<enum>
,ECHOCANCELLATION=<bool>
,SILENCESUPPRESSION=<bool>
,PIC_1=<string>
,PIC_2=<string>
,PIC_3=<string>
,PIC_4=<string>
,FAXSTATISTICS=<bool>
,OUTBANDDTMFSIGNALING=<bool>
,SUPPRESSOUTBANDDTMFWITHG711=<bool>
,QOSCLASSMODE=<enum>
,QOSVALUE=<int>
,QOSPROFILENAME=<string>
,QOSPROFILEBGNAME=<string>
,CODECPRIM=<enum>
,PPPRIM=<int>
,RRPRIM=<enum>
,CODECSEC=<enum>
,PPSEC=<int>
,RRSEC=<enum>
,CODECTHIRD=<enum>
,PPTHIRD=<int>
,RRTHIRD=<enum>
,CODECFOURTH=<enum>
,PPFOURTH=<int>
,RRFOURTH=<enum>
,EXTERNALDNFLAG=<bool>
,COUNTRYCODELEN=<int>
,FEATUREPROFILENAME=<string>
,FEATUREPROFILEBGNAME=<string>
,PHONENAME=<string>
,REGTYPE=<enum>
,RASIPADDR=<string>
,RASPORT=<int>
,TIMETOLIVE=<int>
,ALIASNAME0=<string>
,ALIASTYPE0=<enum>
,ALIASNAME1=<string>
,ALIASTYPE1=<enum>
,ALIASNAME2=<string>
,ALIASTYPE2=<enum>
</CS_H323>
```

24.2.16 Delete Subscriber

```
<RS>
RS - "Delete a subscriber." (Please see WSDL
API:unsp__DeleteSubscriber2)
,SUBSCRIBERDN=<string>
,DELETEGATEWAYOBJECTS=<enum>
</RS>
```

24.2.17 Update Subscriber Directory Number

```
<UD>
UD - "Update Subscriber Directory Number. Change the sub's
DN" (Please see WSDL API:unsp__UpdateSubscriberDn2)
,SUBSCRIBERDN=<string>
,NEWSERVICEID=<string>
,INTERCEPTANNOUNCEMENT=<enum>
</UD>
```

24.2.18 Update Subscriber Capabilities

```
<UC>
UC - "Update Subscriber Capabilities" (Please see WSDL
API:unsp__UpdateSubscriberCapabilities2)
,SUBSCRIBERDN=<string>
,BEARER_SPEECH=<bool>
,BEARER_3_1K=<bool>
,BEARER_56K=<bool>
,BEARER_64K=<bool>
,FAXMODE=<enum>
,ECHOCANCELLATION=<bool>
,SILENCESUPPRESSION=<bool>
,FAXSTATISTICS=<bool>
,OUTBANDDTMFSIGNALING=<bool>
,SUPPRESSOUTBANDDTMFWITHG711=<bool>
</UC>
```

24.2.19 Create Hunt Group

```
<MLHGCREATE>
MLHGCREATE - Create a Mlhg from an existing subscriber.
(Please see WSDL API:unsp__CreateMlhg2)
,BGNAME=<string>
,BOName=<string>
,SUBSCRIBERDN=<string>
,MLHGHUNTTYPE=<enum>
,MLHGEXTERNALDN=<string>
,MLHGOVERFLOWDN=<string>
,MLHGCONFQUEUESIZE=<int>
,MLHGMAXQUEUESIZE=<int>
,MLHGCALLINGPARTYSOURCE=<enum>
,MLHGNOANSADVTIME=<int>

,NOANSADVTIME=<int>
,NOINTERCEPTANNOUNCEMENT=<bool>
,INTERCEPTANNONCEMENTID=<string>
,NIGTHSERVICEDN=<string>
,AUTOMAKEBUSY=<bool>
,MAXTIMEINQUEUE=<int>
,NIGHTSERVICEMANUALSTATUS=<bool>
,POSTCALLTIMER=<int>
,INQUEUEINTERRUPTTIME=<int>
,INQUEUEINTERRUPTINTERCEPT=<string>
```

```
,TMSTATUS=<bool>
,MLHGMAKEBUSYSUBCONTROLLED=<bool>
,MLHGHUNTSUBCONTROLLED=<bool>
,MLHGBUSYSTATUS=<enum>
,MLHGSTOPHUNTSTATUS=<enum>
InQueueInterruptTime=<int>
InQueueInterruptIntercept=<string>
isMlhgRerouteAllowed=<bool>
>SendPilotDisplay=<bool>
isMlhgEnhancedRerouteAllowed=<bool>
OverflowDnNoRerouting=<string>
</MLHGCREATE>
```

24.2.20 Modify Mlhg

```
<MLHGMODIFY>
MLHGMODIFY - Modify Mlhg Information (Please see WSDL
API:unsp__ModifyMlhgInfo2)
,BGNAME=<string>
,BOName=<string>
,MLHGPILOTDN=<string>
,MLHGHUNTTYPE=<enum>
,MLHGEXTERNALDN=<string>
,MLHGOVERFLOWDN=<string>
,MLHGCONFQUEUESIZE=<int>
,MLHGMAXQUEUESIZE=<int>
,MLHGCALLINGPARTYSOURCE=<enum>
,MLHGNOANSADVTIME=<int>
,NOANSADVTIME=<int>
,NOINTERCEPTANNOUNCEMENT=<bool>
,INTERCEPTANNONCEMENTID=<string>
,NIGTHSERVICEDN=<string>
,AUTOMAKEBUSY=<bool>
,MAXTIMEINQUEUE=<int>
,NIGHTSERVICEMANUALSTATUS=<bool>
,POSTCALLTIMER=<int>
,TMSTATUS=<bool>
,INQUEUEINTERRUPTTIME=<int>
,INQUEUEINTERRUPTINTERCEPT=<string>
</MLHGMODIFY>
```

24.2.21 Remove Hunt Group

```
<MLHGREMOVE>
MLHGREMOVE - Remove a Mlhg (Please see WSDL
API:unsp__DeleteMlhg2)
,BGNAME=<string>
,MLHGPILOTDN=<string>
,MLHGREMOVETERMS=<bool>
</MLHGREMOVE>
```

24.2.22 Add Subscriber To Hunt Group

```
<MLHGADDSUB>
MLHGADDSUB - Add Sub To a Multi line Hunting Group (Please
see WSDL API:unsp__AddSubToMlhg2)
,BGNAME=<string>
,MLHGPILOTDN=<string>
,SUBSCRIBERDN=<string>
,POSITION=<int>
,MLHGMAKEBUSYSUBCONTROLLED=<bool>
,MLHGHUNTSUBCONTROLLED=<bool>
,MLHGBUSYSTATUS=<enum>
,MLHGSTOPHUNTSTATUS=<enum>
,MLHGATTENDANT=<bool>
,QUEUEPRIORITY=<int>
</MLHGADDSUB>
```

24.2.23 Modify Hunt Group Terminal

```
<MLHGMODTERM>
MLHGMODTERM -  (Please see WSDL
API:unsp__ModifyMlhgTermInfo2)
,BGNAME=<string>
,BOName=<string>
,MLHGPILOTDN=<string>
,SUBSCRIBERDN=<string>
,MLHGPRIMARYPILOTDN=<string>
,POSITION=<int>
,MLHGMAKEBUSYSUBCONTROLLED=<bool>
,MLHGHUNTSUBCONTROLLED=<bool>
,MLHGBUSYSTATUS=<enum>
,MLHGSTOPHUNTSTATUS=<enum>
,MLHGATTENDANT=<bool>
,QUEUEPRIORITY=<int>
</MLHGMODTERM>
```

24.2.24 Remove Hunt Group Terminal

```
<MLHGREMTERM>
MLHGREMTERM -  (Please see WSDL API:unsp__DeleteMlhgTerm2)
,BGNAME=<string>
,MLHGPILOTDN=<string>
,SUBSCRIBERDN=<string>
</MLHGREMTERM>
```

24.2.25 Create Feature Profile

```
<FPCREATE>
FPCREATE - "Create a Feature Profile" (Please see WSDL
API:unsp__CreateFeatureProfile2)
```

```
,FPNAME=<string>
,BGNAME=<string>
,FEATUREPROFILETYPE=<enum>
,FEATUREPROFILECONNECTIONTYPE=<enum>
</FPCREATE>
```

24.2.26 Delete Feature Profile

```
<FPREMOVE>
FPREMOVE - "Delete a Feature Profile" (Please see WSDL
API:unsp__DeleteFeatureProfile2)
,FPNAME=<string>
,BGNAME=<string>
</FPREMOVE>
```

24.2.27 Update Subscriber Feature Profile

```
<UFP>
UFP - "Update subscribers Feature Profile" (Please see WSDL
API:unsp__UpdateSubscriberFeatureProfile2)
,SUBSCRIBERDN=<string>
,PRESERVELOCPRES=<bool>
,FPNAME=<string>
,BGNAME=<string>
</UFP>
```

24.2.28 Update Subscriber Account Management Info

```
<UM>
UM - Update Subscriber Account Management Info (Please see
WSDL API:unsp__UpdateSubscriberAccountMgtInfo2)
,SUBSCRIBERDN=<string>
,RETAILER=<string>
,ACCOUNTID=<string>
,BILLINGID=<string>
,CLASSOFSERVICE=<string>
,RATEAREA=<string>
,CALLINGLOC=<string>
,TESTLINEFLAG=<bool>
,NUMBERPLANNAME=<string>
,,PRIVACYSUPPORT=<enum>
</UM>
```

24.2.29 Update Account Business Group Information

```
<UMB>
UMB - "Update Account Business Group Information" (Please
```

```
see WSDL API:unsp__UpdateSubscriberBGInfo2)
,SUBSCRIBERDN=<string>
,CPUID=<int>
,BGLINENAME=<string>
,UNICODEBGLINENAME=<string>
,BGATTENDANTNUMBERFLAG=<bool>
,BGEXTENSION=<string>
,BGDEPTNAME=<string>
</UMB>
```

24.2.30 Update Subscriber PIC Information

```
<UP>
UP - "Update Subscriber PIC Information" (Please see WSDL
API:unsp__UpdateSubscriberPICs2)
,SUBSCRIBERDN=<string>
,PIC_1=<string>
,PIC_2=<string>
,PIC_3=<string>
,PIC_4=<string>
</UP>
```

24.2.31 Update Subscriber QoS Profile Information

```
<UQ>
UQ - "Update subscriber QOS Profile Information" (Please see
WSDL API:unsp__UpdateSubscriberQOS2)
,SUBSCRIBERDN=<string>
,QOSPROFILENAME=<string>
,QOSPROFILEBGNAME=<string>
,QOSCLASSMODE=<enum>
,QOSVALUE=<int>
,CODECPRIM=<enum>
,PPPRIM=<int>
,RRPRIM=<enum>
,CODECSEC=<enum>
,PPSEC=<int>
,RRSEC=<enum>
,CODECTHIRD=<enum>
,PPTHIRD=<int>
,RRTHIRD=<enum>
,CODECFOURTH=<enum>
,PPFOURTH=<int>
,RRFOURTH=<enum>
,DQTBRP=<int>
,DQTBSP=<int>
,DQPDRP=<int>
,DQMPUP=<int>
,DQMPSP=<int>
,DQRateP=<int>
,DQSTP=<int>
,DQTBRS=<int>
,DQTBSS=<int>
,DQPDRS=<int>
,DQMPUS=<int>
```

```
,DQMPSS=<int>
,DQRateS=<int>
,DQSTS=<int>
,DQTBRT=<int>
,DQTBST=<int>
,DQPDRT=<int>
,DQMPUT=<int>
,DQMPST=<int>
,DQRateT=<int>
,DQSTT=<int>
,DQTBRF=<int>
,DQTBSF=<int>
,DQPDRF=<int>
,DQMPUF=<int>
,DQMPSF=<int>
,DQRateF=<int>
,DQSTF=<int>
</UQ>
```

24.2.32 Update Subscriber Status

```
<US>
US - "Update subscribers Status" (Please see WSDL
API:unsp__UpdateSubscriberStatus2)
,SUBSCRIBERDN=<string>
,SUBSCRIBERSTATUS=<enum>
</US>
```

24.2.33 Update Subscriber User information

```
<UU>
UU - "Update Subscriber User information" (Please see WSDL
API:unsp__UpdateSubscriberAccountUserInfo2)
,SUBSCRIBERDN=<string>
,DISPLAYNAME=<string>
,UNICODEDISPLAYNAME=<string>
,TIMEZONE=<string>
,PIN=<string>
,PUBLICPIN=<string>
,LANGUAGE=<string>
</UU>
```

24.2.34 Update Connection Information for a SIP Subscriber

```
<UCS>
UCS - "Update Connection Information for a SIP subscriber"
(Please see WSDL API:unsp__UpdateConnectionInfo2)
,SUBSCRIBERDN=<string>
,REGTYPE=<enum>
```

```
,SIGIPADDR=<string>
,SIGPORT=<int>
,LATADIGITS=<string>
,PREREGISTER=<bool>
,IPSEC_REALM=<string>
,IPSEC_UNAME=<string>
,IPSEC_PW=<string>
,IPSEC_SCHEME=<enum>
,IPSEC_SRTP=<enum>
,MAXSIPSESSIONS=<int>
,TRANSPORTTYPE=<enum>
,TRUSTEDSUB=<bool>
,GSMSUB=<bool>
,ASSOCIATEDSIPENDPOINTNAME=<string>
,MEDIAREDIRECTALLOWED=<bool>
,STATICIMSSUB=<bool>
,OVERRIDEIRMCODECRESTRICTIONS=<bool>
,LOOSEROUTINGURI=<string>
,MAXORIGSESSIONLIMIT=<int>
,MAXTERMSESSIONLIMIT=<int>
,AORPERCONTACT=<bool>
,ALIASNAME0=<string>
,ALIASNAME1=<string>
,VIPTYPE=<enum>,
,SENDURITELEPHONESUBFORMAT=<bool>
,SENDINTERNATIONALNUMBERGNF=<bool>
,ANATSUPPORT=<enum>,,RESERVED_2=<bool>
,RESERVED_3=<bool>
,RESERVED_4=<bool>
,DISLONGCALLAUDIT=<bool>
</UCS>
```

24.2.35 Add Contact Information

```
<CTINFOADD>
CTINFOADD - "Add Contact Information" (Please see WSDL
API:unsp__AddContactList2)
,SUBSCRIBERDN=<string>
,CONTACTINFO_LIST_LEN=<int>
,CONTACTHOSTADDR_0=<string>
,CONTACTPORT_0=<int>
,CONTACTCONNECTTYPE_0=<enum>
,CONTACTHOSTADDR_1=<string>
,CONTACTPORT_1=<int>
,CONTACTCONNECTTYPE_1=<enum>
,CONTACTHOSTADDR_2=<string>
,CONTACTPORT_2=<int>
,CONTACTCONNECTTYPE_2=<enum>
,CONTACTHOSTADDR_3=<string>
,CONTACTPORT_3=<int>
,CONTACTCONNECTTYPE_3=<enum>
,CONTACTHOSTADDR_4=<string>
,CONTACTPORT_4=<int>
,CONTACTCONNECTTYPE_4=<enum>
,CONTACTHOSTADDR_5=<string>
,CONTACTPORT_5=<int>
,CONTACTCONNECTTYPE_5=<enum>
,CONTACTHOSTADDR_6=<string>
,CONTACTPORT_6=<int>
```

```
,CONTACTCONNECTTYPE_6=<enum>
,CONTACTHOSTADDR_7=<string>
,CONTACTPORT_7=<int>
,CONTACTCONNECTTYPE_7=<enum>
,CONTACTHOSTADDR_8=<string>
,CONTACTPORT_8=<int>
,CONTACTCONNECTTYPE_8=<enum>
,CONTACTHOSTADDR_9=<string>
,CONTACTPORT_9=<int>
,CONTACTCONNECTTYPE_9=<enum>
,CONTACTHOSTADDR_10=<string>
,CONTACTPORT_10=<int>
,CONTACTCONNECTTYPE_10=<enum>
,CONTACTHOSTADDR_11=<string>
,CONTACTPORT_11=<int>
,CONTACTCONNECTTYPE_11=<enum>
,CONTACTHOSTADDR_12=<string>
,CONTACTPORT_12=<int>
,CONTACTCONNECTTYPE_12=<enum>
,CONTACTHOSTADDR_13=<string>
,CONTACTPORT_13=<int>
,CONTACTCONNECTTYPE_13=<enum>
,CONTACTHOSTADDR_14=<string>
,CONTACTPORT_14=<int>
,CONTACTCONNECTTYPE_14=<enum>
,CONTACTHOSTADDR_15=<string>
,CONTACTPORT_15=<int>
,CONTACTCONNECTTYPE_15=<enum>
,CONTACTHOSTADDR_16=<string>
,CONTACTPORT_16=<int>
,CONTACTCONNECTTYPE_16=<enum>
,CONTACTHOSTADDR_17=<string>
,CONTACTPORT_17=<int>
,CONTACTCONNECTTYPE_17=<enum>
,CONTACTHOSTADDR_18=<string>
,CONTACTPORT_18=<int>
,CONTACTCONNECTTYPE_18=<enum>
,CONTACTHOSTADDR_19=<string>
,CONTACTPORT_19=<int>
,CONTACTCONNECTTYPE_19=<enum>
,CONTACTHOSTADDR_20=<string>
,CONTACTPORT_20=<int>
,CONTACTCONNECTTYPE_20=<enum>
,CONTACTHOSTADDR_21=<string>
,CONTACTPORT_21=<int>
,CONTACTCONNECTTYPE_21=<enum>
,CONTACTHOSTADDR_22=<string>
,CONTACTPORT_22=<int>
,CONTACTCONNECTTYPE_22=<enum>
,CONTACTHOSTADDR_23=<string>
,CONTACTPORT_23=<int>
,CONTACTCONNECTTYPE_23=<enum>
,CONTACTHOSTADDR_24=<string>
,CONTACTPORT_24=<int>
,CONTACTCONNECTTYPE_24=<enum>
,CONTACTHOSTADDR_25=<string>
,CONTACTPORT_25=<int>
,CONTACTCONNECTTYPE_25=<enum>
,CONTACTHOSTADDR_26=<string>
,CONTACTPORT_26=<int>
,CONTACTCONNECTTYPE_26=<enum>
,CONTACTHOSTADDR_27=<string>
```

```
,CONTACTPORT_27=<int>
,CONTACTCONNECTTYPE_27=<enum>
,CONTACTHOSTADDR_28=<string>
,CONTACTPORT_28=<int>
,CONTACTCONNECTTYPE_28=<enum>
,CONTACTHOSTADDR_29=<string>
,CONTACTPORT_29=<int>
,CONTACTCONNECTTYPE_29=<enum>
,CONTACTHOSTADDR_30=<string>
,CONTACTPORT_30=<int>
,CONTACTCONNECTTYPE_30=<enum>
,CONTACTHOSTADDR_31=<string>
,CONTACTPORT_31=<int>
,CONTACTCONNECTTYPE_31=<enum>
,CONTACTHOSTADDR_32=<string>
,CONTACTPORT_32=<int>
,CONTACTCONNECTTYPE_32=<enum>
,CONTACTHOSTADDR_33=<string>
,CONTACTPORT_33=<int>
,CONTACTCONNECTTYPE_33=<enum>
,CONTACTHOSTADDR_34=<string>
,CONTACTPORT_34=<int>
,CONTACTCONNECTTYPE_34=<enum>
,CONTACTHOSTADDR_35=<string>
,CONTACTPORT_35=<int>
,CONTACTCONNECTTYPE_35=<enum>
,CONTACTHOSTADDR_36=<string>
,CONTACTPORT_36=<int>
,CONTACTCONNECTTYPE_36=<enum>
,CONTACTHOSTADDR_37=<string>
,CONTACTPORT_37=<int>
,CONTACTCONNECTTYPE_37=<enum>
,CONTACTHOSTADDR_38=<string>
,CONTACTPORT_38=<int>
,CONTACTCONNECTTYPE_38=<enum>
,CONTACTHOSTADDR_39=<string>
,CONTACTPORT_39=<int>
,CONTACTCONNECTTYPE_39=<enum>
</CTINFOADD>
```

24.2.36 Remove Contact Information

```
<CTINFOREMOVE>
CTINFOREMOVE - "Remove Contact Information" (Please see WSDL
API:unsp__DeleteContactList2)
,SUBSCRIBERDN=<string>
,CONTACTINFO_LIST_LEN=<int>
,CONTACTHOSTADDR_0=<string>
,CONTACTPORT_0=<int>
,CONTACTCONNECTTYPE_0=<enum>
,CONTACTHOSTADDR_1=<string>
,CONTACTPORT_1=<int>
,CONTACTCONNECTTYPE_1=<enum>
,CONTACTHOSTADDR_2=<string>
,CONTACTPORT_2=<int>
,CONTACTCONNECTTYPE_2=<enum>
,CONTACTHOSTADDR_3=<string>
,CONTACTPORT_3=<int>
```

```
,CONTACTCONNECTTYPE_3=<enum>
,CONTACTHOSTADDR_4=<string>
,CONTACTPORT_4=<int>
,CONTACTCONNECTTYPE_4=<enum>
,CONTACTHOSTADDR_5=<string>
,CONTACTPORT_5=<int>
,CONTACTCONNECTTYPE_5=<enum>
,CONTACTHOSTADDR_6=<string>
,CONTACTPORT_6=<int>
,CONTACTCONNECTTYPE_6=<enum>
,CONTACTHOSTADDR_7=<string>
,CONTACTPORT_7=<int>
,CONTACTCONNECTTYPE_7=<enum>
,CONTACTHOSTADDR_8=<string>
,CONTACTPORT_8=<int>
,CONTACTCONNECTTYPE_8=<enum>
,CONTACTHOSTADDR_9=<string>
,CONTACTPORT_9=<int>
,CONTACTCONNECTTYPE_9=<enum>
,CONTACTHOSTADDR_10=<string>
,CONTACTPORT_10=<int>
,CONTACTCONNECTTYPE_10=<enum>
,CONTACTHOSTADDR_11=<string>
,CONTACTPORT_11=<int>
,CONTACTCONNECTTYPE_11=<enum>
,CONTACTHOSTADDR_12=<string>
,CONTACTPORT_12=<int>
,CONTACTCONNECTTYPE_12=<enum>
,CONTACTHOSTADDR_13=<string>
,CONTACTPORT_13=<int>
,CONTACTCONNECTTYPE_13=<enum>
,CONTACTHOSTADDR_14=<string>
,CONTACTPORT_14=<int>
,CONTACTCONNECTTYPE_14=<enum>
,CONTACTHOSTADDR_15=<string>
,CONTACTPORT_15=<int>
,CONTACTCONNECTTYPE_15=<enum>
,CONTACTHOSTADDR_16=<string>
,CONTACTPORT_16=<int>
,CONTACTCONNECTTYPE_16=<enum>
,CONTACTHOSTADDR_17=<string>
,CONTACTPORT_17=<int>
,CONTACTCONNECTTYPE_17=<enum>
,CONTACTHOSTADDR_18=<string>
,CONTACTPORT_18=<int>
,CONTACTCONNECTTYPE_18=<enum>
,CONTACTHOSTADDR_19=<string>
,CONTACTPORT_19=<int>
,CONTACTCONNECTTYPE_19=<enum>
,CONTACTHOSTADDR_20=<string>
,CONTACTPORT_20=<int>
,CONTACTCONNECTTYPE_20=<enum>
,CONTACTHOSTADDR_21=<string>
,CONTACTPORT_21=<int>
,CONTACTCONNECTTYPE_21=<enum>
,CONTACTHOSTADDR_22=<string>
,CONTACTPORT_22=<int>
,CONTACTCONNECTTYPE_22=<enum>
,CONTACTHOSTADDR_23=<string>
,CONTACTPORT_23=<int>
,CONTACTCONNECTTYPE_23=<enum>
,CONTACTHOSTADDR_24=<string>
```

```
,CONTACTPORT_24=<int>
,CONTACTCONNECTTYPE_24=<enum>
,CONTACTHOSTADDR_25=<string>
,CONTACTPORT_25=<int>
,CONTACTCONNECTTYPE_25=<enum>
,CONTACTHOSTADDR_26=<string>
,CONTACTPORT_26=<int>
,CONTACTCONNECTTYPE_26=<enum>
,CONTACTHOSTADDR_27=<string>
,CONTACTPORT_27=<int>
,CONTACTCONNECTTYPE_27=<enum>
,CONTACTHOSTADDR_28=<string>
,CONTACTPORT_28=<int>
,CONTACTCONNECTTYPE_28=<enum>
,CONTACTHOSTADDR_29=<string>
,CONTACTPORT_29=<int>
,CONTACTCONNECTTYPE_29=<enum>
,CONTACTHOSTADDR_30=<string>
,CONTACTPORT_30=<int>
,CONTACTCONNECTTYPE_30=<enum>
,CONTACTHOSTADDR_31=<string>
,CONTACTPORT_31=<int>
,CONTACTCONNECTTYPE_31=<enum>
,CONTACTHOSTADDR_32=<string>
,CONTACTPORT_32=<int>
,CONTACTCONNECTTYPE_32=<enum>
,CONTACTHOSTADDR_33=<string>
,CONTACTPORT_33=<int>
,CONTACTCONNECTTYPE_33=<enum>
,CONTACTHOSTADDR_34=<string>
,CONTACTPORT_34=<int>
,CONTACTCONNECTTYPE_34=<enum>
,CONTACTHOSTADDR_35=<string>
,CONTACTPORT_35=<int>
,CONTACTCONNECTTYPE_35=<enum>
,CONTACTHOSTADDR_36=<string>
,CONTACTPORT_36=<int>
,CONTACTCONNECTTYPE_36=<enum>
,CONTACTHOSTADDR_37=<string>
,CONTACTPORT_37=<int>
,CONTACTCONNECTTYPE_37=<enum>
,CONTACTHOSTADDR_38=<string>
,CONTACTPORT_38=<int>
,CONTACTCONNECTTYPE_38=<enum>
,CONTACTHOSTADDR_39=<string>
,CONTACTPORT_39=<int>
,CONTACTCONNECTTYPE_39=<enum>
</CTINFOREMOVE>
```

24.2.37 Create Business Group

```
<CBG>
CBG - "Create a Business Group" (Please see WSDL
API:unsp__CreateBG2)
,BGNAME=<string>
,NUMBERPLANNAME=<string>
,DISPLAYNUMBER=<string>
,DNRESERVEENABLED=<bool>
```

```
</CBG>
```

24.2.38 Delete Business Group

```
<RBG>
RBG - "Delete a Business Group" (Please see WSDL
API:unsp__DeleteBG2)
,BGNAME=<string>
</RBG>
```

24.2.39 Update Business Group Status

```
<UBS>
UBS - "Update Business Group Status" (Please see WSDL
API:unsp__UpdateBGStatus2)
,BGNAME=<string>
,BGSTATUS=<enum>
</UBS>
```

24.2.40 Add Business Group Main Number

```
<ABM>
ABM - "Add Main number to a Business Group" (Please see WSDL
API:unsp__AddBGMainNumber2)
,BGNAME=<string>
,BGMAINNUMBER=<string>
,BGMAINNUMBERATTENDANT=<string>
,AUTOATTENDANTNUMBER=<string>
,AUTOATTENDANTSTATUS=<enum>
</ABM>
```

24.2.41 Delete Business Group Main Number

```
<RBM>
RBM - "Delete BG Main number" (Please see WSDL
API:unsp__DeleteBGMainNumber2)
,BGNAME=<string>
,BGMAINNUMBER=<string>
</RBM>

Update Business Group Main Number
<UBM>
UBM - "Update Main Number of a Business Group" (Please see
WSDL API:unsp__UpdateBGMainNumber2)
,BGNAME=<string>
,BGMAINNUMBER=<string>
,BGMAINNUMBERATTENDANT=<string>
```

```
,AUTOATTENDANTNUMBER=<string>
,AUTOATTENDANTSTATUS=<enum>
</UBM>
```

24.2.42 Update Business Group MDR Information

```
<UBMDR>
UBMDR - "Update MDR Information" (Please see WSDL
API:unsp__UpdateBGParms2)
,BGNAME=<string>
,MDRINDICATOR=<enum>
,MDRCUSTID=<string>
</UBMDR>
```

24.2.43 Update Business Group Media Server Information

```
<UBCNT>
UBCNT - "Update Media server related Information" (Please
see WSDL API:unsp__UpdateBGParms2)
,BGNAME=<string>
,MEDIASVRRETRYCOUNT=<int>
,MEDIASVRTIMEOUTINTERVAL=<int>
,INPUTRETRYCOUNT=<int>
</UBCNT>
```

24.2.44 Update Subscriber Keyset Info

```
<UK>
UK - Update Subscriber Keyset Info (Please see WSDL
API:unsp__UpdateKeysetInfo2)
,SUBSCRIBERDN=<string>
,KEYSETUSE=<enum>
,KEYSETMANUALHOLDTIMER=<int>
,CALLFORWARDLEDINDICATION=<bool>
,KEYSETLINEAPPEARANCEENTRIES_LIST_LEN=<int>
,KEYSETLINEAPPEARANCE_0=<string>
,ORIGKEYSETOPERATIONMODE_0=<enum>
,TERMKEYSETOPERATIONMODE_0=<enum>
,KEYSETLINEAPPEARANCE_1=<string>
,ORIGKEYSETOPERATIONMODE_1=<enum>
,TERMKEYSETOPERATIONMODE_1=<enum>
,KEYSETLINEAPPEARANCE_2=<string>
,ORIGKEYSETOPERATIONMODE_2=<enum>
,TERMKEYSETOPERATIONMODE_2=<enum>
,KEYSETLINEAPPEARANCE_3=<string>
,ORIGKEYSETOPERATIONMODE_3=<enum>
,TERMKEYSETOPERATIONMODE_3=<enum>
,KEYSETLINEAPPEARANCE_4=<string>
,ORIGKEYSETOPERATIONMODE_4=<enum>
```

```
,TERMKEYSETOPERATIONMODE_4=<enum>
,KEYSETLINEAPPEARANCE_5=<string>
,ORIGKEYSETOPERATIONMODE_5=<enum>
,TERMKEYSETOPERATIONMODE_5=<enum>
,KEYSETLINEAPPEARANCE_6=<string>
,ORIGKEYSETOPERATIONMODE_6=<enum>
,TERMKEYSETOPERATIONMODE_6=<enum>
,KEYSETLINEAPPEARANCE_7=<string>
,ORIGKEYSETOPERATIONMODE_7=<enum>
,TERMKEYSETOPERATIONMODE_7=<enum>
,KEYSETLINEAPPEARANCE_8=<string>
,ORIGKEYSETOPERATIONMODE_8=<enum>
,TERMKEYSETOPERATIONMODE_8=<enum>
,KEYSETLINEAPPEARANCE_9=<string>
,ORIGKEYSETOPERATIONMODE_9=<enum>
,TERMKEYSETOPERATIONMODE_9=<enum>
,KEYSETLINEAPPEARANCE_10=<string>
,ORIGKEYSETOPERATIONMODE_10=<enum>
,TERMKEYSETOPERATIONMODE_10=<enum>
,KEYSETLINEAPPEARANCE_11=<string>
,ORIGKEYSETOPERATIONMODE_11=<enum>
,TERMKEYSETOPERATIONMODE_11=<enum>
,KEYSETLINEAPPEARANCE_12=<string>
,ORIGKEYSETOPERATIONMODE_12=<enum>
,TERMKEYSETOPERATIONMODE_12=<enum>
,KEYSETLINEAPPEARANCE_13=<string>
,ORIGKEYSETOPERATIONMODE_13=<enum>
,TERMKEYSETOPERATIONMODE_13=<enum>
,KEYSETLINEAPPEARANCE_14=<string>
,ORIGKEYSETOPERATIONMODE_14=<enum>
,TERMKEYSETOPERATIONMODE_14=<enum>
,KEYSETLINEAPPEARANCE_15=<string>
,ORIGKEYSETOPERATIONMODE_15=<enum>
,TERMKEYSETOPERATIONMODE_15=<enum>
,KEYSETLINEAPPEARANCE_16=<string>
,ORIGKEYSETOPERATIONMODE_16=<enum>
,TERMKEYSETOPERATIONMODE_16=<enum>
,KEYSETLINEAPPEARANCE_17=<string>
,ORIGKEYSETOPERATIONMODE_17=<enum>
,TERMKEYSETOPERATIONMODE_17=<enum>
,KEYSETLINEAPPEARANCE_18=<string>
,ORIGKEYSETOPERATIONMODE_18=<enum>
,TERMKEYSETOPERATIONMODE_18=<enum>
,KEYSETLINEAPPEARANCE_19=<string>
,ORIGKEYSETOPERATIONMODE_19=<enum>
,TERMKEYSETOPERATIONMODE_19=<enum>
,KEYSETLINEAPPEARANCE_20=<string>
,ORIGKEYSETOPERATIONMODE_20=<enum>
,TERMKEYSETOPERATIONMODE_20=<enum>
,KEYSETLINEAPPEARANCE_21=<string>
,ORIGKEYSETOPERATIONMODE_21=<enum>
,TERMKEYSETOPERATIONMODE_21=<enum>
,KEYSETLINEAPPEARANCE_22=<string>
,ORIGKEYSETOPERATIONMODE_22=<enum>
,TERMKEYSETOPERATIONMODE_22=<enum>
,KEYSETLINEAPPEARANCE_23=<string>
,ORIGKEYSETOPERATIONMODE_23=<enum>
,TERMKEYSETOPERATIONMODE_23=<enum>
,KEYSETLINEAPPEARANCE_24=<string>
,ORIGKEYSETOPERATIONMODE_24=<enum>
,TERMKEYSETOPERATIONMODE_24=<enum>
,KEYSETLINEAPPEARANCE_25=<string>
```

```
,ORIGKEYSETOPERATIONMODE_25=<enum>
,TERMKEYSETOPERATIONMODE_25=<enum>
,KEYSETLINEAPPEARANCE_26=<string>
,ORIGKEYSETOPERATIONMODE_26=<enum>
,TERMKEYSETOPERATIONMODE_26=<enum>
,KEYSETLINEAPPEARANCE_27=<string>
,ORIGKEYSETOPERATIONMODE_27=<enum>
,TERMKEYSETOPERATIONMODE_27=<enum>
,KEYSETLINEAPPEARANCE_28=<string>
,ORIGKEYSETOPERATIONMODE_28=<enum>
,TERMKEYSETOPERATIONMODE_28=<enum>
,KEYSETLINEAPPEARANCE_29=<string>
,ORIGKEYSETOPERATIONMODE_29=<enum>
,TERMKEYSETOPERATIONMODE_29=<enum>
</UK>
```

24.2.45 Create Business Group Department

```
<BGCREATEDEPT>
BGCREATEDEPT - "Create a Business Group department" (Please
see WSDL API:unsp__CreateBGDept2)
,BGNAME=<string>
,BGDEPTNAME=<string>
</BGCREATEDEPT>
```

24.2.46 Delete Business Group Department

```
<BGDELETEDEPT>
BGDELETEDEPT - "Delete a Business Group department" (Please
see WSDL API:unsp__DeleteBGDept2)
,BGNAME=<string>
,BGDEPTNAME=<string>
</BGDELETEDEPT>
```

24.2.47 Create Business Group E9-1-1 Subnet

```
<E911BGSUBNETCREATE>
E911BGSUBNETCREATE - "Create a Business Group E911 subnet"
(Please see WSDL API:unsp__CreateBgSubnet2)
,BGNAME=<string>
,BOName=<string>
,SUBNETIP=<string>
,SUBNETMASK=<string>
,LOCATIONDOMAIN=<string>
,SCANRATE=<int>
,SCANRESPONSETIMEOUT=<int>
,DESCRIPTION=<string>
,LINLIST_E911LIN=<string>
,LINLIST_CBN=<string>
,LINLIST_DCBN=<string>
,LINLIST_USEDCBN=<int>
```

```
,ROUTENUMBER=<string>
,PASSCODE=<string>
,E911ALIDBID=<int>
</E911BGSUBNETCREATE>
```

24.2.48 Delete Business Group E9-1-1 Subnet

```
<E911BGSUBNETDELETE>
E911BGSUBNETDELETE - "Delete a Business Group E911 subnet"
(Please see WSDL API:unsp__DeleteBgSubnet2)
,BGNAME=<string>
,SUBNETIP=<string>
,SUBNETMASK=<string>
,LOCATIONDOMAIN=<string>
,LINLIST_E911LIN=<string>
,LINLIST_CBN=<string>
,LINLIST_DCBN=<string>
,LINLIST_USEDCBN=<int>
</E911BGSUBNETDELETE>
```

When specifying no LINs, then BG subnet and all LINs of the specified BG Subnet are removed.

```
<E911BGSUBNETDELETE>
E911BGSUBNETDELETE - "Delete a Business Group E911 subnet"
(Please see WSDL API:unsp__DeleteBgSubnet2)
,BGNAME=<string>
,SUBNETIP=<string>
,SUBNETMASK=<string>
</E911BGSUBNETDELETE>
```

24.2.49 Update Business Group E9-1-1 Subnet

```
<E911BGSUBNETUPDATE>
E911BGSUBNETUPDATE - "Update a Business Group E911 subnet"
(Please see WSDL API:unsp__UpdateBgSubnet2)
,BGNAME=<string>
,BOName=<string>
,SUBNETIP=<string>
,SUBNETMASK=<string>
,LOCATIONDOMAIN=<string>
,SCANRATE=<int>
,SCANRESPONSETIMEOUT=<int>
,DESCRIPTION=<string>
,LINLIST_E911LIN=<string>
,LINLIST_CBN=<string>
,LINLIST_DCBN=<string>
,LINLIST_USEDCBN=<int>
,ROUTENUMBER=<string>
,PASSCODE=<string>
,E911ALIDBID=<int>
,SENDLININSTEADCPN=<int>
</E911BGSUBNETUPDATE>
```

24.2.50 Create Endpoint Profile

```
<EPCREATE>
EPCREATE - "Create an endpoint profile" (Please see WSDL
API:unsp__CreateEpProfile2)
,EPNAME=<string>
,EPTYPE=<enum>
,BGNAME=<string>
,CLASSOFSERVICE=<string>
,RATEAREA=<string>
,LATADIGITS=<string>
,PRIVACYSUPPORT=<enum>
,LANGUAGE=<string>
,CALLINGLOC=<string>
,FAILEDCALLSINTERCEPTTREATMENT=<bool>
,NUMBERPLANNAME=<string>
,PIC_1=<string>
,PIC_2=<string>
,PIC_3=<string>
,PIC_4=<string>
,BEARER_SPEECH=<bool>
,BEARER_3_1K=<bool>
,BEARER_56K=<bool>
,BEARER_64K=<bool>
,FAXMODE=<enum>
,ECHOCANCELLATION=<bool>
,SILENCESUPPRESSION=<bool>
,FAXSTATISTICS=<bool>
,OUTBANDDTMFSIGNALING=<bool>
,SUPPRESSOUTBANDDTMFWITHG711=<bool>
,QOSCLASSMODE=<enum>
,QOSVALUE=<int>
,QOSPROFILENAME=<string>
,QOSPROFILEBGNAME=<string>
,CODECPRIM=<enum>
,PPPRIM=<int>
,TOSPRIM=<int>
,CODECSEC=<enum>
,PPSEC=<int>
,TOSSEC=<int>
,CODECTHIRD=<enum>
,PPTHIRD=<int>
,TOSTHIRD=<int>
,CODECFOURTH=<enum>
,PPFOURTH=<int>
,TOSFOURTH=<int>
</EPCREATE>
```

24.2.51 Delete Endpoint Profile

```
<EPDEL>
EPDEL - "Delete an endpoint profile" (Please see WSDL
API:unsp__DeleteEpProfile2)
,EPNAME=<string>
```

```
</EPDEL>

Update Endpoint Profile Capabilities
<EPUPDATECAP>
EPUPDATECAP - "Update an endpoint profile Capabilites"
(Please see WSDL API:unsp__UpdateEpCapabilities2)
,EPNAME=<string>
,BEARER_SPEECH=<bool>
,BEARER_3_1K=<bool>
,BEARER_56K=<bool>
,BEARER_64K=<bool>
,FAXMODE=<enum>
,ECHOCANCELLATION=<bool>
,SILENCESUPPRESSION=<bool>
,FAXSTATISTICS=<bool>
,OUTBANDDTMFSIGNALING=<bool>
,SUPPRESSOUTBANDDTMFWITHG711=<bool>
</EPUPDATECAP>
```

24.2.52 Update Endpoint Profile Management Information

```
<EPUPDATEMGT>
EPUPDATEMGT - "Update endpoint profile management
information" (Please see WSDL API:unsp__UpdateEpMgtInfo2)
,EPNAME=<string>
,BGNAME=<string>
,CLASSOFSERVICE=<string>
,RATEAREA=<string>
,LATADIGITS=<string>
,PRIVACYSUPPORT=<enum>
,LANGUAGE=<string>
,CALLINGLOC=<string>
,FAILEDCALLSINTERCEPTTREATMENT=<bool>
,NUMBERPLANNAME=<string>
,IMPACTLEVEL=<enum>
</EPUPDATEMGT>
```

24.2.53 Update Endpoint Profile PICs

```
<EPUPDATEPICS>
EPUPDATEPICS - "Update Endpoint profiles PICS" (Please see
WSDL API:unsp__UpdateEpPics2)
,EPNAME=<string>
,PIC_1=<string>
,PIC_2=<string>
,PIC_3=<string>
,PIC_4=<string>
</EPUPDATEPICS>
```

24.2.54 Update Endpoint Profile QoS Information

```
<EPUPDATEQOS>
EPUPDATEQOS - "Update endpoint profile QOS" (Please see WSDL
API:unsp__UpdateEpQos2)
,EPNAME=<string>
,QOSCLASSMODE=<enum>
,QOSVALUE=<int>
,QOSPROFILENAME=<string>
,QOSPROFILEBGNAME=<string>
,CODECPRIM=<enum>
,PPPRIM=<int>
,TOSPRIM=<int>
,CODECSEC=<enum>
,PPSEC=<int>
,TOSSEC=<int>
,CODECTHIRD=<enum>
,PPTHIRD=<int>
,TOSTHIRD=<int>
,CODECFOURTH=<enum>
,PPFOURTH=<int>
,TOSFOURTH=<int>
</EPUPDATEQOS>
```

24.2.55 Create QoS Profile

```
<QOSPROFILECREATE>
QOSPROFILECREATE -  (Please see WSDL
API:unsp__CreateQOSProfile2)
,QOSPROFILENAME=<string>
,QOSPROFILEBGNAME=<string>
,QOSCLASSMODE=<enum>
,QOSVALUE=<int>
,CODECPRIM=<enum>
,PPPRIM=<int>
,TOSPRIM=<int>
,DQTBRP=<int>
,DQTBSP=<int>
,DQPDRP=<int>
,DQMPUP=<int>
,DQMPSP=<int>
,DQRateP=<int>
,DQSTP=<int>
,CODECSEC=<enum>
,PPSEC=<int>
,TOSSEC=<int>
,DQTBRS=<int>
,DQTBSS=<int>
,DQPDRS=<int>
,DQMPUS=<int>
,DQMPSS=<int>
,DQRateS=<int>
,DQSTS=<int>
,CODECTHIRD=<enum>
,PPTHIRD=<int>
,TOSTHIRD=<int>
,DQTBRT=<int>
```

```
,DQTBST=<int>
,DQPDRT=<int>
,DQMPUT=<int>
,DQMPST=<int>
,DQRateT=<int>
,DQSTT=<int>
,CODECFOURTH=<enum>
,PPFOURTH=<int>
,TOSFOURTH=<int>
,DQTBRF=<int>
,DQTBSF=<int>
,DQPDRF=<int>
,DQMPUF=<int>
,DQMPSF=<int>
,DQRateF=<int>
,DQSTF=<int>
,RSRCMGMTTYP=<enum>
,PCMMCLNTTYP=<enum>
,PCMMPS=<string>
,PCMMVOL=<int>
,PCMMVOLEXPMTHD=<enum>
,PCMMTIMR=<int>
,PCMMTIMREXPMTHD=<enum>
</QOSPROFILECREATE>
```

24.2.56 Modify QoS Profile

```
<QOSPROFILEMODIFY>
QOSPROFILEMODIFY -  (Please see WSDL
API:unsp__ModifyQOSProfile2)
,QOSPROFILENAME=<string>
,QOSPROFILEBGNAME=<string>
,QOSCLASSMODE=<enum>
,QOSVALUE=<int>
,CODECPRIM=<enum>
,PPPRIM=<int>
,TOSPRIM=<int>
,RRPRIM=<enum>
,CODECSEC=<enum>
,PPSEC=<int>
,TOSSEC=<int>
,RRSEC=<enum>
,CODECTHRD=<enum>
,PPTHRD=<int>
,TOSTHRD=<int>
,RRTHRD=<enum>
,CODECFOURTH=<enum>
,PPFOURTH=<int>
,TOSFOURTH=<int>
,RRFOURTH=<enum>
,DQTBRP=<int>
,DQTBSP=<int>
,DQPDRP=<int>
,DQMPUP=<int>
,DQMPSP=<int>
,DQRateP=<int>
,DQSTP=<int>
,DQTBRS=<int>
```

```
,DQTBSS=<int>
,DQPDRS=<int>
,DQMPUS=<int>
,DQMPSS=<int>
,DQRateS=<int>
,DQSTS=<int>
,DQTBRT=<int>
,DQTBST=<int>
,DQPDRT=<int>
,DQMPUT=<int>
,DQMPST=<int>
,DQRateT=<int>
,DQSTT=<int>
,DQTBRF=<int>
,DQTBSF=<int>
,DQPDRF=<int>
,DQMPUF=<int>
,DQMPSF=<int>
,DQRateF=<int>
,DQSTF=<int>
,RSRCMGMTTYP=<enum>
,PCMMCLNTTYP=<enum>
,PCMMPS=<string>
,PCMMVOL=<int>
,PCMMVOLEXPMTHD=<enum>
,PCMMTIMR=<int>
,PCMMTIMREXPMTHD=<enum>
</QOSPROFILEMODIFY>
```

24.2.57 Delete QoS Profile

```
<QOSPROFILEDEL>
QOSPROFILEDEL - Delete a QOS Profile (Please see WSDL
API:unsp__DeleteQOSProfile2)
,QOSPROFILENAME=<string>
,QOSPROFILEBGNAME=<string>
</QOSPROFILEDEL>
```

24.2.58 Create Destination Code

```
<DESTCODECREATE>
DESTCODECREATE - "Creates a Destination Code" (Please see
WSDL API:unsp__CreateDestCode2 in Provisioning Doc for
Attribute details)
,DESTINATIONCODE=<string>
,CODEINDEXNAME=<string>
,NUMPA=<string>
,HOMEDNOFFICECODE=<string>
,NUMBERPLANNAME=<string>
,DESTINATIONNAME=<string>
,RATEAREA=<string>
,CLASSOFSERVICE=<string>
,DESTINATIONTYPE=<enum>
,TRAFFICTYPENAME=<string>
,NATUREOFADDRESS=<enum>
```

```
,OPERPREFIXNOTALLOWED=<bool>
,NATIONALPREFIXNOTALLOWED=<bool>
,NATIONPREFIXREQUIRED=<bool>
,CACNOTALLOWED=<bool>
,INTRIGPROCREQUIRED=<bool>
,DESTNEWCODE=<string>
,DESTCOUNTRYCODE=<string>
</DESTCODECREATE>
```

24.2.59 Modify Destination Code

```
<DESTCODEMODIFY>
DESTCODEMODIFY - "Modifies a Destination Code" (Please see
WSDL API:unsp__ModifyDestCode2 in Provisioning Doc for
Attribute details)
,DESTINATIONCODE=<string>
,CODEINDEXNAME=<string>
,NUMPA=<string>
,HOMEDNOFFICECODE=<string>
,NUMBERPLANNAME=<string>
,DESTINATIONNAME=<string>
,RATEAREA=<string>
,CLASSOFSERVICE=<string>
,DESTINATIONTYPE=<enum>
,TRAFFICTYPENAME=<string>
,NATUREOFADDRESS=<enum>
,OPERPREFIXNOTALLOWED=<bool>
,NATIONALPREFIXNOTALLOWED=<bool>
,NATIONPREFIXREQUIRED=<bool>
,CACNOTALLOWED=<bool>
,INTRIGPROCREQUIRED=<bool>
,DESTNEWCODE=<string>
,DESTCOUNTRYCODE=<string>
</DESTCODEMODIFY>
```

24.2.60 Remove Destination Code

```
<DESTCODEREMOVE>
DESTCODEREMOVE - Remove a Destination Code (Please see WSDL
API:unsp__DeleteDestCode2)
,DESTINATIONCODE=<string>
,CODEINDEXNAME=<string>
,NUMBERPLANNAME=<string>
,RATEAREA=<string>
,CLASSOFSERVICE=<string>
,NATUREOFADDRESS=<enum>
</DESTCODEREMOVE>
```

24.2.61 Create Number Plan

```
<NPCREATE>
```

```
NPCREATE - Creates a Number Plan (Please see WSDL
API:unsp__CreateNumberPlan2)
,NUMBERPLANNAME=<string>
,NUMBERPLANID=<int>
,BGNAME=<string>
</NPCREATE>
```

24.2.62 Remove Number Plan

```
<NPREMOVE>
NPREMOVE - Removes a Number Plan (Please see WSDL
API:unsp__DeleteNumberPlan2)
,NUMBERPLANNAME=<string>
</NPREMOVE>
```

24.2.63 Create Authorization Code

```
<AUTHCODECREATE>
AUTHCODECREATE - The authorization code to be used must be
between 2 and 14 characters. (Please see WSDL
API:unsp__CreateAuthCode)
,BGNAME=<string>
,AUTHCODETYPE=enum
,SUBSCRIBERDN=string
,COUNTAUTHCODELIST=<int>
,AUTHCODE_0=<string>
,AUTHCODE_1=<string>
,AUTHCODE_2=<string>
,AUTHCODE_3=<string>
,AUTHCODE_4=<string>
,AUTHCODE_5=<string>
,AUTHCODE_6=<string>
,AUTHCODE_7=<string>
,AUTHCODE_8=<string>
,AUTHCODE_9=<string>
,AUTHCODE_10=<string>
,AUTHCODE_11=<string>
,AUTHCODE_12=<string>
,AUTHCODE_13=<string>
,AUTHCODE_14=<string>
,AUTHCODE_15=<string>
,AUTHCODE_16=<string>
,AUTHCODE_17=<string>
,AUTHCODE_18=<string>
,AUTHCODE_19=<string>
</AUTHCODECREATE>
```

24.2.64 Delete Authorization Code

```
<AUTHCODEDEL>
AUTHCODEDEL - Delete Authorization Code (Please see WSDL
```

```
API:unsp__DeleteAuthCode)
,BGNAME=<string>
,COUNTAUTHCODELIST=<int>
,AUTHCODE_0=<string>
,AUTHCODE_1=<string>
,AUTHCODE_2=<string>
,AUTHCODE_3=<string>
,AUTHCODE_4=<string>
,AUTHCODE_5=<string>
,AUTHCODE_6=<string>
,AUTHCODE_7=<string>
,AUTHCODE_8=<string>
,AUTHCODE_9=<string>
,AUTHCODE_10=<string>
,AUTHCODE_11=<string>
,AUTHCODE_12=<string>
,AUTHCODE_13=<string>
,AUTHCODE_14=<string>
,AUTHCODE_15=<string>
,AUTHCODE_16=<string>
,AUTHCODE_17=<string>
,AUTHCODE_18=<string>
,AUTHCODE_19=<string>
</AUTHCODEDEL>
```

24.2.65 Create Prefix Access Code

```
<APAC>
APAC - Create Prefix Access Code (Please see WSDL
API:unsp__CreatePrefixAccessCode2)
,NUMBERPLANNAME=<string>
,DIGITS=<string>
,DNMINLEN=<int>
,DNMAXLEN=<int>
,PREFIXTYPE=<enum>
,DIGITPOS=<int>
,DIGITSINSERT=<string>
,TRAILINGDIGITPOS=<int>
,NOA=<enum>
,DESTTYPE=<enum>
,DESTNAME=<string>
,CACLENGTH=<int>
</APAC>
```

24.2.66 Create PNP Location Code

```
<APNPLOC>
APNPLOC - Create PNP Location Code (Please see WSDL
API:unsp__CreatePnpLocationCode2)
,NUMBERPLANNAME=<string>
,LOCCODE=<string>
,LOCL0LEN=<int>
,LOCL1LEN=<int>
,LOCL2LEN=<int>
```

```
,LOCSKIPDIGITS=<int>
</APNPLOC>
```

24.2.67 Update PNP Location Code

```
<UPNPLOC>
UPNPLOC - Modify Update PNP Location Code (Please see WSDL
API:unsp__ModifyPnpLocationCode2)
,NUMBERPLANNAME=<string>
,LOCCODE=<string>
,LOCL0LEN=<int>
,LOCL1LEN=<int>
,LOCL2LEN=<int>
,LOCSKIPDIGITS=<int>
</UPNPLOC>
```

24.2.68 Remove PNP Location Code

```
<RPNPLOC>
RPNPLOC - Remove PNP Location Code (Please see WSDL
API:unsp__DeletePnpLocationCode2)
,NUMBERPLANNAME=<string>
,LOCCODE=<string>
</RPNPLOC>
```

24.2.69 Update Prefix Access Code

```
<UPAC>
UPAC - Update Modify Prefix Access Code (Please see WSDL
API:unsp__ModifyPrefixAccessCode2)
,NUMBERPLANNAME=<string>
,DIGITS=<string>
,DNMINLEN=<int>
,DNMAXLEN=<int>
,PREFIXTYPE=<enum>
,DIGITPOS=<int>
,DIGITSINSERT=<string>
,TRAILINGDIGITPOS=<int>
,NOA=<enum>
,DESTTYPE=<enum>
,DESTNAME=<string>
</UPAC>
```

24.2.70 Remove Prefix Access Code

```
<RPAC>
RPAC - Remove Prefix Access Code (Please see WSDL
```

```
API:unsp__DeletePrefixAccessCode2)
,NUMBERPLANNAME=<string>
,DIGITS=<string>
,DNMINLEN=<int>
,DNMAXLEN=<int>
</RPAC>
```

24.2.71 Create PNP Extension

```
<APNPEXT>
APNPEXT - Create PNP Extension (Please see WSDL
API:unsp__CreatePnpExtension2)
,NUMBERPLANNAME=<string>
,EXTPREFIX=<string>
,EXTLEN=<int>
,LOCCODE=<string>
,E164PREFIX=<string>
,DESTTYPE=<enum>
,DESTNAME=<string>
</APNPEXT>
```

24.2.72 Update PNP Extension

```
<UPNPEXT>
UPNPEXT - Update Modify PNP Extension (Please see WSDL
API:unsp__ModifyPnpExtension2)
,NUMBERPLANNAME=<string>
,EXTPREFIX=<string>
,EXTLEN=<int>
,LOCCODE=<string>
,E164PREFIX=<string>
,DESTTYPE=<enum>
,DESTNAME=<string>
</UPNPEXT>
```

24.2.73 Remove PNP Extension

```
<RPNPEXT>
RPNPEXT - Remove PNP Extension (Please see WSDL
API:unsp__DeletePnpExtension2)
,NUMBERPLANNAME=<string>
,EXTPREFIX=<string>
</RPNPEXT>
```

24.2.74 Create Display Number Definition

```
<DNDCREATE>
DNDCREATE - "Create Display Number Definition" (Please see
```

```
WSDL API:unsp__CreateDNDefinition)
,NUMBERPLANNAME=<string>
,COUNTRYCODE=<string>
,AREACODE=<string>
,EXCHANGECODE=<string>
,E164SKIPDIGIT=<int>
,LOCATIONCODE=<string>
,LOCL2LEN=<int>
,LOCL1LEN=<int>
,LOCL0LEN=<int>
,PNPSKIPDIGIT=<int>
</DNDCREATE>
```

24.2.75 Delete Display Number Definition

```
<DNDDELETE>
DNDDELETE - "Delete Display Number Definition" (Please see
WSDL API:unsp__CreateDNDefinition)
,NUMBERPLANNAME=<string>
,COUNTRYCODE=<string>
,AREACODE=<string>
,EXCHANGECODE=<string>
,E164SKIPDIGIT=<int>
,LOCATIONCODE=<string>
,LOCL2LEN=<int>
,LOCL1LEN=<int>
,LOCL0LEN=<int>
,PNPSKIPDIGIT=<int>
</DNDDELETE>
```

24.2.76 Create Display Number Prefix

```
<DNPCREATE>
DNPCREATE - "Create Display Number Prefix" (Please see WSDL
API:unsp__CreateDNPrefix)
,NUMBERPLANNAME=<string>
,DNPCALLNUMBERPLANINDICATOR=<enum>
,DNPCALLTYPEOFNUMBER=<enum>
,NETWORKACCESSCODE=<string>
,NUMBERPREFIXCODE=<string>
</DNPCREATE>
```

24.2.77 Modify Display Number Prefix

```
<DNPMODIFY>
DNPMODIFY - "Modify Display Number Prefix" (Please see WSDL
API:unsp__ModifyDNPrefix)
,NUMBERPLANNAME=<string>
,DNPCALLNUMBERPLANINDICATOR=<enum>
,DNPCALLTYPEOFNUMBER=<enum>
,NETWORKACCESSCODE=<string>
```

```
,NUMBERPREFIXCODE=<string>
</DNPMODIFY>
```

24.2.78 Delete Display Number Prefix

```
<DNPDELETE>
DNPDELETE - "Delete Display Number Prefix" (Please see WSDL
API:unsp__DeleteDNPrefix)
,NUMBERPLANNAME=<string>
,DNPCALLNUMBERPLANINDICATOR=<enum>
,DNPCALLTYPEOFNUMBER=<enum>
</DNPDELETE>
```

24.2.79 Create Display Number Modification

```
<DNMCREATE>
DNMCREATE - "Create Display Number Modification" (Please see
WSDL API:unsp__CreateDNModification)
,FROMNUMBERPLANNAME=<string>
,TONUMBERPLANNAME=<string>
,DNMTYPEOFNUMBERIN=<enum>
,DNMTYPEOFNUMBER=<enum>
,PRIVACYTYPE=<enum>
,OPTIMIZED=<bool>
</DNMCREATE>
```

24.2.80 Modify Display Number Modification

```
<DNMMODIFY>
DNMMODIFY - "Modify Display Number Modification" (Please see
WSDL API:unsp__ModifyDNModification)
,FROMNUMBERPLANNAME=<string>
,TONUMBERPLANNAME=<string>
,DNMTYPEOFNUMBERIN=<enum>
,DNMTYPEOFNUMBER=<enum>
,PRIVACYTYPE=<enum>
,OPTIMIZED=<bool>
</DNMMODIFY>
```

24.2.81 Delete Display Number Modification

```
<DNMDELETE>
DNMDELETE - "Delete Display Number Modification" (Please see
WSDL API:unsp__DeleteDNModification)
,FROMNUMBERPLANNAME=<string>
,TONUMBERPLANNAME=<string>
,DNMTYPEOFNUMBERIN=<enum>
```

```
</DNMDELETE>
```

24.2.82 Update Business Group Common Numbering Plan

```
<BGPARAMSMODIFY>
BGPARAMSMODIFY - "Update BG Parameters,
BGCommonNumberingPlan"
 (Please see WSDL API:unsp__UpdateBGParms2)
,BGNAME=<string>
,BGCOMMONNUMBERPLANNAME=<string>
</BGPARAMSMODIFY>
```

24.2.83 Update Business Group Default QoS profile

```
<UBQOSP>
UBQOSP - "Update default QOS profile" (Please see WSDL
API:unsp__UpdateBGParms2)
,BGNAME=<string>
,QOSPROFILENAME=<string>
,QOSPROFILEBGNAME=<string>
</UBQOSP>
```

24.2.84 Create CAC Group

```
<CCG>
CCG - "Create a CAC Group" (Please see WSDL
API:unsp__CreateCACGroup)
,BUSINESSGROUPNAME=<string>
,BRANCHOFFICENAME=<string>
,CACGROUPNAME=<string>
,CACGROUPTYPE=<enum>
,ROUTERIPADDRESS=<string>
,INTERFACENAME=<string>
,ACCESSLINKTYPE=<enum>
,ROUTERIPADDRESS2=<string>
,INTERFACENAME2=<string>
,ACCESSLINKTYPE2=<enum>
,NUMCACGROUPADDRESS=<int>
,CACGROUPADDRESS0=<string>
,CACGROUPADDRESS1=<string>
,CACGROUPADDRESS2=<string>
,CACGROUPADDRESS3=<string>
,CACGROUPADDRESS4=<string>
,CACGROUPADDRESS5=<string>
,CACGROUPADDRESS6=<string>
,CACGROUPADDRESS7=<string>
,CACGROUPADDRESS8=<string>
,CACGROUPADDRESS9=<string>
,CACGROUPADDRESS10=<string>
```

```
,CACGROUPADDRESS11=<string>
,CACGROUPADDRESS12=<string>
,CACGROUPADDRESS13=<string>
,CACGROUPADDRESS14=<string>
,CACGROUPADDRESS15=<string>
,CACGROUPADDRESS16=<string>
,CACGROUPADDRESS17=<string>
,CACGROUPADDRESS18=<string>
,CACGROUPADDRESS19=<string>
,CACGROUPADDRESS20=<string>
,CACGROUPADDRESS21=<string>
,CACGROUPADDRESS22=<string>
,CACGROUPADDRESS23=<string>
,CACGROUPADDRESS24=<string>
,CACGROUPADDRESS25=<string>
,CACGROUPADDRESS26=<string>
,CACGROUPADDRESS27=<string>
,CACGROUPADDRESS28=<string>
,CACGROUPADDRESS29=<string>
,CACGROUPADDRESS30=<string>
,CACGROUPADDRESS31=<string>
,CACGROUPADDRESS32=<string>
,CACGROUPADDRESS33=<string>
,CACGROUPADDRESS34=<string>
,CACGROUPADDRESS35=<string>
,CACGROUPADDRESS36=<string>
,CACGROUPADDRESS37=<string>
,CACGROUPADDRESS38=<string>
,CACGROUPADDRESS39=<string>
,CACGROUPADDRESS40=<string>
,CACGROUPADDRESS41=<string>
,CACGROUPADDRESS42=<string>
,CACGROUPADDRESS43=<string>
,CACGROUPADDRESS44=<string>
,CACGROUPADDRESS45=<string>
,CACGROUPADDRESS46=<string>
,CACGROUPADDRESS47=<string>
,CACGROUPADDRESS48=<string>
,CACGROUPADDRESS49=<string>
,CACGROUPADDRESS50=<string>
,CACGROUPADDRESS51=<string>
,CACGROUPADDRESS52=<string>
,CACGROUPADDRESS53=<string>
,CACGROUPADDRESS54=<string>
,CACGROUPADDRESS55=<string>
,CACGROUPADDRESS56=<string>
,CACGROUPADDRESS57=<string>
,CACGROUPADDRESS58=<string>
,CACGROUPADDRESS59=<string>
,CACGROUPADDRESS60=<string>
,CACGROUPADDRESS61=<string>
,CACGROUPADDRESS62=<string>
,CACGROUPADDRESS63=<string>
</CCG>
```

24.2.85 Delete CAC Group

```
<DCG>
```

```
DCG - "Delete a CAC Group" (Please see WSDL
API:unsp__DeleteCACGroup)
,CACGROUPNAME=<string>
</DCG>
```

24.2.86 Remove CAC Group Address

```
<DCGA>
DCGA - "Removes one or more address from an existing CAC
Group" (Please see WSDL API:unsp__RemoveCACGroupAddr)
,CACGROUPNAME=<string>
,NUMCACGROUPADDRESS=<int>
,CACGROUPADDRESS0=<string>
,CACGROUPADDRESS1=<string>
,CACGROUPADDRESS2=<string>
,CACGROUPADDRESS3=<string>
,CACGROUPADDRESS4=<string>
,CACGROUPADDRESS5=<string>
,CACGROUPADDRESS6=<string>
,CACGROUPADDRESS7=<string>
,CACGROUPADDRESS8=<string>
,CACGROUPADDRESS9=<string>
,CACGROUPADDRESS10=<string>
,CACGROUPADDRESS11=<string>
,CACGROUPADDRESS12=<string>
,CACGROUPADDRESS13=<string>
,CACGROUPADDRESS14=<string>
,CACGROUPADDRESS15=<string>
,CACGROUPADDRESS16=<string>
,CACGROUPADDRESS17=<string>
,CACGROUPADDRESS18=<string>
,CACGROUPADDRESS19=<string>
,CACGROUPADDRESS20=<string>
,CACGROUPADDRESS21=<string>
,CACGROUPADDRESS22=<string>
,CACGROUPADDRESS23=<string>
,CACGROUPADDRESS24=<string>
,CACGROUPADDRESS25=<string>
,CACGROUPADDRESS26=<string>
,CACGROUPADDRESS27=<string>
,CACGROUPADDRESS28=<string>
,CACGROUPADDRESS29=<string>
,CACGROUPADDRESS30=<string>
,CACGROUPADDRESS31=<string>
,CACGROUPADDRESS32=<string>
,CACGROUPADDRESS33=<string>
,CACGROUPADDRESS34=<string>
,CACGROUPADDRESS35=<string>
,CACGROUPADDRESS36=<string>
,CACGROUPADDRESS37=<string>
,CACGROUPADDRESS38=<string>
,CACGROUPADDRESS39=<string>
,CACGROUPADDRESS40=<string>
,CACGROUPADDRESS41=<string>
,CACGROUPADDRESS42=<string>
,CACGROUPADDRESS43=<string>
,CACGROUPADDRESS44=<string>
,CACGROUPADDRESS45=<string>
```

```
,CACGROUPADDRESS46=<string>
,CACGROUPADDRESS47=<string>
,CACGROUPADDRESS48=<string>
,CACGROUPADDRESS49=<string>
,CACGROUPADDRESS50=<string>
,CACGROUPADDRESS51=<string>
,CACGROUPADDRESS52=<string>
,CACGROUPADDRESS53=<string>
,CACGROUPADDRESS54=<string>
,CACGROUPADDRESS55=<string>
,CACGROUPADDRESS56=<string>
,CACGROUPADDRESS57=<string>
,CACGROUPADDRESS58=<string>
,CACGROUPADDRESS59=<string>
,CACGROUPADDRESS60=<string>
,CACGROUPADDRESS61=<string>
,CACGROUPADDRESS62=<string>
,CACGROUPADDRESS63=<string>
</DCGA>
```

24.2.87 Add CAC Group Address

```
<CCGA>
CCGA - "Adds one or more addresses to an existing CAC Group"
(Please see WSDL API:unsp__AddCACGroupAddr)
,CACGROUPNAME=<string>
,NUMCACGROUPADDRESS=<int>
,CACGROUPADDRESS0=<string>
,CACGROUPADDRESS1=<string>
,CACGROUPADDRESS2=<string>
,CACGROUPADDRESS3=<string>
,CACGROUPADDRESS4=<string>
,CACGROUPADDRESS5=<string>
,CACGROUPADDRESS6=<string>
,CACGROUPADDRESS7=<string>
,CACGROUPADDRESS8=<string>
,CACGROUPADDRESS9=<string>
,CACGROUPADDRESS10=<string>
,CACGROUPADDRESS11=<string>
,CACGROUPADDRESS12=<string>
,CACGROUPADDRESS13=<string>
,CACGROUPADDRESS14=<string>
,CACGROUPADDRESS15=<string>
,CACGROUPADDRESS16=<string>
,CACGROUPADDRESS17=<string>
,CACGROUPADDRESS18=<string>
,CACGROUPADDRESS19=<string>
,CACGROUPADDRESS20=<string>
,CACGROUPADDRESS21=<string>
,CACGROUPADDRESS22=<string>
,CACGROUPADDRESS23=<string>
,CACGROUPADDRESS24=<string>
,CACGROUPADDRESS25=<string>
,CACGROUPADDRESS26=<string>
,CACGROUPADDRESS27=<string>
,CACGROUPADDRESS28=<string>
,CACGROUPADDRESS29=<string>
,CACGROUPADDRESS30=<string>
```

```
,CACGROUPADDRESS31=<string>
,CACGROUPADDRESS32=<string>
,CACGROUPADDRESS33=<string>
,CACGROUPADDRESS34=<string>
,CACGROUPADDRESS35=<string>
,CACGROUPADDRESS36=<string>
,CACGROUPADDRESS37=<string>
,CACGROUPADDRESS38=<string>
,CACGROUPADDRESS39=<string>
,CACGROUPADDRESS40=<string>
,CACGROUPADDRESS41=<string>
,CACGROUPADDRESS42=<string>
,CACGROUPADDRESS43=<string>
,CACGROUPADDRESS44=<string>
,CACGROUPADDRESS45=<string>
,CACGROUPADDRESS46=<string>
,CACGROUPADDRESS47=<string>
,CACGROUPADDRESS48=<string>
,CACGROUPADDRESS49=<string>
,CACGROUPADDRESS50=<string>
,CACGROUPADDRESS51=<string>
,CACGROUPADDRESS52=<string>
,CACGROUPADDRESS53=<string>
,CACGROUPADDRESS54=<string>
,CACGROUPADDRESS55=<string>
,CACGROUPADDRESS56=<string>
,CACGROUPADDRESS57=<string>
,CACGROUPADDRESS58=<string>
,CACGROUPADDRESS59=<string>
,CACGROUPADDRESS60=<string>
,CACGROUPADDRESS61=<string>
,CACGROUPADDRESS62=<string>
,CACGROUPADDRESS63=<string>
</CCGA>
```

24.2.88 Create CAC Policy

```
<CCP>
CCP - "Create a CAC Policy" (Please see WSDL
API:unsp__CreateCACPolicy)
,CACGROUPNAME=<string>
,CACPOLICYTYPE=<int>
,CACPOLICYLIMITTYPE=<enum>
,CACPOLICYOPTIONFLAG=<enum>
,CALLLIMIT=<int>
,BANDWIDTHLIMIT=<int>
,SECONDARYCALLLIMIT=<int>
,SECONDARYBANDWIDTHLIMIT=<int>
,CACPOLICYALLOWALLCODEC=<bool>
,CACPOLICYALLOWG711ALAWCODEC=<bool>
,CACPOLICYALLOWG711ULAWCODEC=<bool>
,CACPOLICYALLOWG722CODEC=<bool>
,CACPOLICYALLOWG7231CODEC=<bool>
,CACPOLICYALLOWG72616CODEC=<bool>
,CACPOLICYALLOWG72624CODEC=<bool>
,CACPOLICYALLOWG72632CODEC=<bool>
,CACPOLICYALLOWG72640CODEC=<bool>
,CACPOLICYALLOWG728CODEC=<bool>
```

```
,CACPOLICYALLOWG729CODEC=<bool>
,CACPOLICYALLOWILBCCODEC=<bool>
,CACPOLICYALLOWAMRCODEC=<bool>
,CACPOLICYALLOWAMRWBCODEC=<bool>
CACPOLICYALLOWAACLCCODEC=<bool>
CACPOLICYALLOWG722124CODEC=<bool>
CACPOLICYALLOWG722132CODEC=<bool>
CACPOLICYALLOWG722148CODEC=<bool>
CACPOLICYALLOWALLVIDEOCODEC=<bool>
CACPOLICYALLOWH263CODEC=<bool>
CACPOLICYALLOWH264CODEC=<bool>
MAXPROFILELEVEL=<enum>
</CCP>
```

24.2.89 Delete CAC Policy

```
<DCP>
DCP - "Delete a CAC Policy" (Please see WSDL
API:unsp__DeleteCACPolicy)
,CACGROUPNAME=<string>
,CACPOLICYTYPE=<int>
</DCP>
```

24.2.90 Create Group To Group CAC Policy

```
<CGTGCP>
CGTGCP - "Creates a Group-To-Group CAC Policy" (Please see
WSDL API:unsp__CreateGToGCACPolicy in Provisioning Doc for
Attribute details)
,CACGROUP1NAME=<string>
,CACGROUP2NAME=<string>
,CACPOLICYTYPE=<int>
,CACPOLICYLIMITTYPE=<enum>
,CACPOLICYOPTIONFLAG=<enum>
,CALLLIMIT=<int>
,BANDWIDTHLIMIT=<int>
,CACPOLICYALLOWALLCODEC=<bool>
,CACPOLICYALLOWG711ALAWCODEC=<bool>
,CACPOLICYALLOWG711ULAWCODEC=<bool>
,CACPOLICYALLOWG722CODEC=<bool>
,CACPOLICYALLOWG7231CODEC=<bool>
,CACPOLICYALLOWG72616CODEC=<bool>
,CACPOLICYALLOWG72624CODEC=<bool>
,CACPOLICYALLOWG72632CODEC=<bool>
,CACPOLICYALLOWG72640CODEC=<bool>
,CACPOLICYALLOWG728CODEC=<bool>
,CACPOLICYALLOWG729CODEC=<bool>
,CACPOLICYALLOWILBCCODEC=<bool>
,CACPOLICYALLOWAMRCODEC=<bool>
,CACPOLICYALLOWAMRWBCODEC=<bool>
CACPOLICYALLOWAACLCCODEC=<bool>
CACPOLICYALLOWG722124CODEC=<bool>
CACPOLICYALLOWG722132CODEC=<bool>
CACPOLICYALLOWG722148CODEC=<bool>
CACPOLICYALLOWH263CODEC=<bool>
CACPOLICYALLOWH264CODEC=<bool>
```

```
MAXPROFILELEVEL=<enum>
</CGTGCP>
```

24.2.91 Delete Group To Group CAC Policy

```
<DGTGCP>
DGTGCP - "Deletes a Group-To-Group CAC Policy" (Please see
WSDL API:unsp__DeleteGToGCACPolicy in Provisioning Doc for
Attribute details)
,CACGROUP1NAME=<string>
,CACGROUP2NAME=<string>
,CACPOLICYTYPE=<int>
,=<>
</DGTGCP>
```

24.2.92 Create Business Group Speed Calling List

```
<BGSPEEDDIALLISTCREATE>
BGSPEEDDIALLISTCREATE - "Creation of BG Speed Calling Lists"
(Please see WSDL API:unsp__CreateBgSpeedDialList)
,BGNAME=<string>
,LISTID=<int>
,LISTNAME=<string>
,BGSD_ENTRYID=<int>
,BGSD_ENTRYDATA=<string>
</BGSPEEDDIALLISTCREATE>
```

24.2.93 Update Business Group Speed Calling List

```
<BGSPEEDDIALLISTUPDATE>
BGSPEEDDIALLISTUPDATE - "Update the ListName of a List"
(Please see WSDL API:unsp__UpdateBgSpeedDialList)
,BGNAME=<string>
,LISTID=<int>
,LISTNAME=<string>
</BGSPEEDDIALLISTUPDATE>
```

24.2.94 Delete Business Group Speed Calling List

```
<BGSPEEDDIALLISTDELETE>
BGSPEEDDIALLISTDELETE - "Delete an entire Speed Dial List"
(Please see WSDL API:unsp__DeleteBgSpeedDialList)
,BGNAME=<string>
,LISTID=<int>
</BGSPEEDDIALLISTDELETE>
```

24.2.95 Delete Business Group Speed Calling List Entries

```
<BGSPEEDDIALLISTENTRIESDELETE>
BGSPEEDDIALLISTENTRIESDELETE - "Delete the Entries of a
SPDial List (Please see WSDL
API:unsp__DeleteBgSpeedDialListEntries)
,BGNAME=<string>
,LISTID=<int>
,BGSD_ENTRYID=<int>
</BGSPEEDDIALLISTENTRIESDELETE>
```

24.2.96 Update Business Group Speed Calling List Entries

```
<BGSPEEDDIALLISTENTRIESUPDATE>
BGSPEEDDIALLISTENTRIESUPDATE - "Udpate the Entries of a
List" (Please see WSDL
API:unsp__UpdateBgSpeedDialListEntries)
,BGNAME=<string>
,LISTID=<int>
,BGSD_ENTRYID=<int>
,BGSD_ENTRYDATA=<string>
</BGSPEEDDIALLISTENTRIESUPDATE>
```

24.2.97 Assign Day Schedule

```
<AssignDaySchedule>
AssignDayScheduleRequest(Please see WSDL
API:unsp__assignDaySchedule for more details)
AssignDaySchedule
   ,TimeDestinationName=<string>
   ,DayScheduleNameSun=<string>
   ,DayScheduleNameMon=<string>
   ,DayScheduleNameTue=<string>
   ,DayScheduleNameWed=<string>
   ,DayScheduleNameThu=<string>
   ,DayScheduleNameFri=<string>
   ,DayScheduleNameSat=<string>
<\AssignDaySchedule>
```

24.2.98 Assign Treatment

```
<AssignTreatment>
TreatmentDataInfo(Please see WSDL API:unsp__assignTreatment
for more details)
AssignTreatment
   ,Cycle=<long>
```

```
   ,Interval=<long>
   ,Type=<enum unsp__TreatmentType>
   ,SipReasonId=<long>
   ,TreatmentInfoName=<string>
   ,TreatmentInfoOrder=<enum unsp__TreatmentOrder>
   ,InterceptInfoName=<string>
   ,ToneName=<string>
   ,ToneDuration=<long>
   ,ToneDestinationName=<string>
   ,ToneDestinationCircuit=<string>
   ,ToneDestinationType=<enum
unsp__InterceptDestinationType>
   ,AnnounceName=<string>
   ,AnnounceAnswer=<enum unsp__YesNo>
   ,AnnounceBargeIn=<enum unsp__YesNo>
   ,AnnounceDestinationName=<string>
   ,AnnounceDestinationCircuit=<string>
   ,AnnounceDestinationType=<enum
unsp__InterceptDestinationType>
<\AssignTreatment>
```

24.2.99 Block Media Gateway

```
<BlockMediaGateway>
BlockMediaGatewayRequest(Please see WSDL
API:unsp__blockMediaGateway for more details)
BlockMediaGateway
   ,GatewayName=<string>
   ,AdminStatus=<enum unsp__ResourceStatus>
<\BlockMediaGateway>
```

24.2.100 Block Trunk

```
<BlockTrunk>
BlockTrunkRequest(Please see WSDL API:unsp__blockTrunk for
more details)
BlockTrunk
   ,TrunkGroupName=<RtmTrunkGroupName>
   ,TrunkGroupNumber=<unsignedShort>
   ,TrunkNumberMin=<int>
   ,TrunkNumberMax=<int>
   ,AdminStatus=<enum unsp__ResourceStatus>
<\BlockTrunk>
```

24.2.101 Block Trunk Group

```
<BlockTrunkGroup>
BlockTrunkGroupRequest(Please see WSDL
API:unsp__blockTrunkGroup for more details)
BlockTrunkGroup
   ,TrunkGroupName=<RtmTrunkGroupName>
```

```
   ,TrunkGroupNumber=<unsignedInt>
   ,AdminStatus=<enum unsp__ResourceStatus>
<\BlockTrunkGroup>
```

24.2.102 Create Alias

```
<Alias>
CreateAliasRequest(Please see WSDL API:unsp__createAlias
for more details)
Alias
   ,NodeName=<string>
   ,NumberAliases=<short>
   ,AliasesNodeName=<string>
   ,AliasesAliasName=<string>
   ,AliasesAliasType=<enum unsp__AliasType>
   ,AliasesRegistration=<short>
<\Alias>
```

24.2.103 Create Calling Location

```
<CallingLocation>
CreateCallingLocationRequest(Please see WSDL
API:unsp__createCallingLocation for more details)
CallingLocation
   ,Code=<string>
   ,Name=<string>
<\CallingLocation>
```

24.2.104 Create Class Of Service

```
<ClassService>
CreateClassServiceRequest(Please see WSDL
API:unsp__createClassService for more details)
ClassService
   ,Name=<string>
   ,Number=<long>
   ,DefaultCpcCos=<long>
   ,DefaultCpcCosName=<string>
<\ClassService>
```

24.2.105 Create Code Process

```
<CodeProcess>
CreateCodeProcessRequest(Please see WSDL
API:unsp__createCodeProcess for more details)
CodeProcess
```

```
   ,NumberOfOperations=<short>
   ,CodeProcessCodeProcessName=<string>
   ,CodeProcessNatureOfAddress=<enum
unsp__E164NatureOfAddr>
   ,CodeProcessRetranslate=<bool>
   ,CodeProcessDestinationName=<string>
   ,CodeProcessDestinationType=<enum unsp__DestCodeDt>
   ,OperationListDigitPosition=<long>
   ,OperationListDigitsToInsert=<string>
   ,OperationListDigitsToDelete=<long>
<\CodeProcess>
```

24.2.106 Create Day Schedule

```
<DaySchedule>
CreateDayScheduleRequest(Please see WSDL
API:unsp__createDaySchedule for more details)
DaySchedule
   ,Name=<string>
   ,TimeDestinationName=<string>
<\DaySchedule>
```

24.2.107 Create Destination

```
<Destination>
CreateDestinationRequest(Please see WSDL
API:unsp__createDestination for more details)
Destination
   ,NumberRoutes=<long>
   ,DestinationName=<string>
   ,DestinationSignaling=<enum unsp__SignalingType>
   ,DestinationBearer=<enum unsp__BearerCaps>
   ,DestinationDestinationAttributes=<enum
unsp__DestinationAttributes>
   ,DestinationAttributes=<enum unsp__Attributes>
   ,DestinationNumberPlanName=<string>
   ,DestinationPrefixAreaCode=<enum unsp__Attributes>
   ,DestinationPrefixCountryCode=<enum unsp__Attributes>
   ,DestinationReRoute=<enum unsp__Attributes>
   ,RouteListId=<long>
   ,RouteListName=<string>
   ,RouteListReferenceName=<string>
   ,RouteListReferenceType=<enum unsp__ReferenceType>
   ,RouteListDigitsToDelete=<short>
   ,RouteListDigitsToInsert=<string>
   ,RouteListNatureOfAddress=<enum unsp__E164NatureOfAddr>
   ,RouteListCacSignaling=<enum unsp__CacSignaling>
<\Destination>
```

24.2.108 Create Endpoint

```
<Endpoint>
```

```
CreateEndpointRequest(Please see WSDL
API:unsp__createEndpoint for more details)
Endpoint
   ,NumberAliases=<short>
   ,EndpointName=<string>
   ,EndpointIpAddress=<string>
   ,EndpointAltIpAddress=<string>
   ,EndpointRasIpAddress=<string>
,EndpointManagementIpAddress=<string>
   ,EndpointPort=<int>
   ,EndpointAltPort=<int>
   ,EndpointRasPort=<int>
   ,EndpointEndpointType=<enum unsp__EndpointType>
   ,EndpointRegistration=<short>
   ,EndpointTimeToLive=<double>
   ,EndpointAccountAuth=<enum unsp__AccountAuth>
   ,EndpointProfileName=<string>
,EndpointSessions=<long>
,EndpointMaxOriginatingSessionLimit=<long>
   ,EndpointMaxTerminatingSessionLimit=<long>
,EndpointTransport=<enum unsp__TransportType>
   ,EndpointAttributes=<long>
   ,EndpointAddressOfRecord=<string>
   ,EndpointServerFailover=<enum unsp__EpFailover>
   ,EndpointKeysetUse=<enum unsp__KeysetUse>
   ,EndpointCountryCode=<string>
   ,EndpointAssociatedEndpointName=<string>
   ,EndpointIvtgNameDomain=<string>
   ,EndpointAttributes=<long>
,EndpointLooseRoutingURI=<string>
,EnablePrack=<bool>
,EndpointSrtpSupport=<enum unsp__SrtpSupport>
,EndpointAnatSupport=<enum unsp__Anat>
,EndpointLocationDomain=<string>
,SetNPIandTONtoUnknown=<bool>
,SendXferingNrIsoCgNrForXferedCalls=<bool>
,IncludeRestrNrsInFromHeader=<bool>
,EnableSessionTimer=<bool>
,EndpointUseEndpointDefaultHomeDnAsAuthNumber=<bool>
,EndpointUseSubscriberHomeDnAsAuthNumber=<bool>
,EndpointDisableLongCallAudit=<bool>
,EndpointReserved2=<bool>
,EndpointReserved3=<bool>
,EndpointReserved4=<bool>
,SendReceiveImpactLevel=<bool>
,AddressOfRecord=<string>
,SupportCallbackPathReservation=<bool>
,SendProgStopCallProcTimer=<bool>
,VoiceMail=<bool>
,EnablePrack=<bool>
,SrtpSupport=<unsp__SrtpSupport
,LocationDomain=<string>
,AliasesNodeName=<string>
   ,AliasesAliasName=<string>
   ,AliasesAliasType=<enum unsp__AliasType>
   ,AliasesCustomerName=<string>
   ,AliasesRegistration=<short>
<\Endpoint>
```

24.2.109 Create HomeDN

```
<HomeDn>
CreateHomeDnRequest(Please see WSDL API:unsp__createHomeDn
for more details)
HomeDn
   ,StartCode=<string>
   ,EndCode=<string>
   ,OfficeCode=<string>
   ,<\HomeDn>
```

24.2.110 Create Intercept

```
<Intercept>
InterceptRequest(Please see WSDL API:unsp__createIntercept
for more details)
Intercept
   ,CreateName=<boolean>
   ,NumberTreatments=<short>
   ,InterceptCycle=<long>
   ,InterceptInterval=<long>
   ,InterceptIsDefault=<enum unsp__IsDefault>
   ,InterceptDefaultName=<string>
   ,InterceptInfoName=<string>
   ,TreatmentCycle=<long>
   ,TreatmentInterval=<long>
   ,TreatmentType=<enum unsp__TreatmentType>
   ,TreatmentSipReasonId=<long>
   ,TreatmentInfoName=<string>
   ,TreatmentInfoOrder=<enum unsp__TreatmentOrder>
   ,InterceptInfoName=<string>
   ,ToneName=<string>
   ,ToneDuration=<long>
   ,ToneDestinationName=<string>
   ,ToneDestinationCircuit=<string>
   ,ToneDestinationType=<enum
unsp__InterceptDestinationType>
   ,AnnounceName=<string>
   ,AnnounceAnswer=<enum unsp__YesNo>
   ,AnnounceBargeIn=<enum unsp__YesNo>
   ,AnnounceDestinationName=<string>
   ,AnnounceDestinationCircuit=<string>
   ,AnnounceDestinationType=<enum
unsp__InterceptDestinationType>
<\Intercept>
```

24.2.111 Create Media Gateway Circuits

```
<MediaGatewayCircuits>
CreateMediaGatewayCircuitsRequest(Please see WSDL
API:unsp__createMediaGatewayCircuits for more details)
MediaGatewayCircuits
   ,GatewayName=<string>
   ,CircuitType=<enum unsp__CircuitAssignment>
   ,StartCircuitId=<string>
```

```
   ,EndCircuitId=<string>
<\MediaGatewayCircuits>
```

24.2.112 Create Media Gateway

```
<MediaGateway>
CreateMediaGatewayRequest(Please see WSDL
API:unsp__createMediaGateway for more details)
MediaGateway
   ,GatewayName=<string>
   ,GatewayProtocolType=<enum unsp__GatewayProtocolType>
   ,GatewayProtocolVersion=<enum
unsp__GatewayProtocolVersion>
   ,MgCircuitFormat=<string>
   ,NetworkTypes=<int>
   ,CodecTypes=<int>
   ,MgAddrCallp=<string>
   ,MgAddrProvision=<string>
   ,Timeout=<unsignedInt>
   ,RetryCount=<unsignedInt>
   ,IpAddrAllocMethod=<enum unsp__IPAddressAllocMethod>
   ,Fqdn=<string>
   ,LocalClli=<string>
   ,GwAssignMethod=<enum unsp__GatewayAssignmentMethod>
   ,MediaServerVersion=<enum unsp__MSVersion>
   ,TimerHist=<int>
   ,TimerMax=<int>
   ,TransmissionTimerLen=<int>
   ,FaxT38Relay=<enum unsp__T38FaxRelay>
   ,DtmfRelay=<enum unsp__DtmfRelay>
   ,MultiHomingFlag=<enum unsp__MultiHoming>
,OverloadSupport=<enum unsp__YesNo>
,OverloadAllowTwCalls=<enum unsp__YesNo>
   ,OverloadReturnCodeOne=<int>
   ,OverloadReturnCodeTwo=<int>
   ,OverloadReturnCodeThree=<int>
   ,OverloadGapIntervalOne=<int>
   ,OverloadGapIntervalTwo=<int>
   ,OverloadGapIntervalThree=<int>
,LocationDomain=<string>
<\MediaGateway>
```

24.2.113 Create Office Code

```
<OfficeCode>
CreateOfficeCodeRequest(Please see WSDL
API:unsp__createOfficeCode for more details)
OfficeCode
   ,OfficeCode=<string>
   ,CountryCodeLength=<short>
   ,AreaCodeLength=<short>
   ,Attributes=<enum unsp__Attributes>
   ,ChargeArea=<string>
<\OfficeCode>
```

24.2.114 Create Origin Destination

```
<OriginDestination>
OriginDestinationRequest(Please see WSDL
API:unsp__createOriginDestination for more details)
OriginDestination
   ,Name=<string>
   ,Type=<enum unsp__OrigDestType>
<\OriginDestination>
```

24.2.115 Create Origin Route

```
<OriginRoute>
CreateOriginRoute(Please see WSDL
API:unsp__createOriginRoute for more details)
OriginRoute
   ,OriginDestinationName=<string>
   ,RateAreaName=<string>
   ,DestinationName=<string>
   ,DestinationType=<enum unsp__OriginRouteDestinationType>
   ,IsupCircuitCode=<long>
<\OriginRoute>
```

24.2.116 Create Period Schedule

```
<PeriodSchedule>
CreatePeriodScheduleRequest(Please see WSDL
API:unsp__createPeriodSchedule for more details)
PeriodSchedule
   ,Name=<string>
   ,DayScheduleName=<string>
   ,TimeDestinationName=<string>
   ,StartTime=<string>
   ,StopTime=<string>
   ,DestinationName=<string>
<\PeriodSchedule>
```

24.2.117 Create Rate Area

```
<RateArea>
CreateRateAreaRequest(Please see WSDL
API:unsp__createRateArea for more details)
RateArea
   ,Group=<long>
   ,Name=<string>
   ,NpaNxx=<string>
   ,SlhrName=<string>
   ,PointCode=<long>
   ,NodeVariant=<enum unsp__NodeVariant>
   ,Lata=<string>
```

```
<\RateArea>
```

24.2.118 Create Route

```
<Route>
CreateRouteRequest(Please see WSDL API:unsp__createRoute
for more details)
Route
   ,NumRoutes=<long>
   ,GenericRouteDestinationName=<string>
   ,GenericRouteDestinationSignaling=<enum
unsp__SignalingType>
   ,GenericRouteDestinationBearer=<enum unsp__BearerCaps>
   ,GenericRouteDestinationDestinationAttributes=<enum
unsp__DestinationAttributes>
   ,GenericRouteDestinationAttributes=<enum
unsp__Attributes>
   ,GenericRouteDestinationNumberPlanName=<string>
   ,GenericRouteDestinationPrefixAreaCode=<enum
unsp__Attributes>
   ,GenericRouteDestinationPrefixCountryCode=<enum
unsp__Attributes>
   ,GenericRouteDestinationReRoute=<enum unsp__Attributes>
   ,RouteListId=<long>
   ,RouteListName=<string>
   ,RouteListReferenceName=<string>
   ,RouteListReferenceType=<enum unsp__ReferenceType>
   ,RouteListDigitsToDelete=<short>
   ,RouteListDigitsToInsert=<string>
   ,RouteListNatureOfAddress=<enum unsp__E164NatureOfAddr>
   ,RouteListCacSignaling=<enum unsp__CacSignaling>
<\Route>
```

24.2.119 Create Time Destination

```
<TimeDestination>
TimeDestinationRequest(Please see WSDL
API:unsp__createTimeDestination for more details)
TimeDestination
   ,Name=<string>
<\TimeDestination>
```

24.2.120 Create Trunk

```
<Trunk>
CreateTrunkRequest(Please see WSDL API:unsp__createTrunk
for more details)
Trunk
   ,TrunkGroupName=<RtmTrunkGroupName>
   ,TrunkGroupNumber=<unsignedShort>
   ,TrunkNumberMin=<int>
   ,CicNumberMin=<int>
```

```
   ,MediaGatewayName=<string>
   ,MediaCicMinId=<string>
   ,MediaCicMaxId=<string>
   ,AdminStatus=<enum unsp__ResourceStatus>
   ,HandlingGroupNumber=<short>
<\Trunk>
```

24.2.121 Create Trunk Group

```
<TrunkGroup>
CreateTrunkGroupRequest(Please see WSDL
API:unsp__createTrunkGroup for more details)
TrunkGroup
   ,TrunkGroupName=<RtmTrunkGroupName>
   ,TrunkGroupNumber=<unsignedInt>
   ,AdminStatus=<enum unsp__ResourceStatus>
   ,TrunkType=<enum unsp__TrunkType>
   ,TrunkGroupComment=<RtmLongString>
   ,SignalProfileName=<string>
   ,FarEndDpc=<RtmPointCode>
   ,OwnPointCodeType=<enum unsp__OPCType>
   ,OwnPointCode=<RtmPointCode>
   ,Usage=<enum unsp__TrunkGroupUsage>
   ,Owner=<RtmLongString>
   ,RemoteClli=<string>
   ,CDRRecordingIns=<enum unsp__CDRRecordingIns>
   ,SearchMethod=<enum unsp__SearchMethod>
   ,GainControl=<int>
   ,CotProcedure=<enum unsp__COTProcedure>
   ,CotFrequency=<int>
   ,CirCarrierIndicator=<enum
unsp__CircuitCarrierIndicator>
   ,AlarmCarrierIndicator=<enum
unsp__AlarmCarrierIndicator>
   ,HopCountInitValue=<int>
   ,AccessTandem=<enum unsp__YesNo>
   ,LataDigits=<RtmLataDigit>
   ,DefaultDn=<RtmDefaultDn>
   ,RateAreaName=<RtmRateAreaName>
   ,ClassOfServiceName=<RtmClassOfServceName>
   ,NetworkIndicator=<enum unsp__NetworkIndicator>
   ,SigTranIndicator=<enum unsp__YesNo>
   ,PreferredTrunkUsage=<int>
   ,BothwayTrunksReserve=<int>
   ,PriorityTrunksReserve=<int>
   ,FarEndCarrierIdCode=<RtmFarEndCarrierIdCode>
   ,FarEndCarrierSwitchName=<RtmFarEndCarrierSwName>
   ,POIflag=<enum unsp__POIFlag>
   ,POIstage=<enum unsp__POIStage>
   ,POIChargeArea=<RtmPOIChargeArea>
   ,Language=<string>
<\TrunkGroup>
```

24.2.122 Modify Class Of Service

```
<ModifyClassService>
ModifyClassServiceRequest(Please see WSDL
API:unsp__modifyClassService for more details)
ModifyClassService
   ,Name=<string>
   ,DefaultCpcCosName=<string>
<\ModifyClassService>
```

24.2.123 Modify Code Process

```
<ModifyCodeProcess>
ModifyCodeProcessRequest(Please see WSDL
API:unsp__modifyCodeProcess for more details)
ModifyCodeProcess
   ,NumberOfOperations=<short>
   ,CodeProcessCodeProcessName=<string>
   ,CodeProcessNatureOfAddress=<enum
unsp__E164NatureOfAddr>
   ,CodeProcessRetranslate=<bool>
   ,CodeProcessDestinationName=<string>
   ,CodeProcessDestinationType=<enum unsp__DestCodeDt>
   ,OperationListDigitPosition=<long>
   ,OperationListDigitsToInsert=<string>
   ,OperationListDigitsToDelete=<long>
<\ModifyCodeProcess>
```

24.2.124 Modify Day Schedule

```
<ModifyDaySchedule>
ModifyDayScheduleRequest(Please see WSDL
API:unsp__modifyDaySchedule for more details)
ModifyDaySchedule
   ,ModifyType=<enum unsp__DnModifyType>
   ,TimeDestinationName=<string>
   ,DayScheduleNameSun=<string>
   ,DayScheduleNameMon=<string>
   ,DayScheduleNameTue=<string>
   ,DayScheduleNameWed=<string>
   ,DayScheduleNameThu=<string>
   ,DayScheduleNameFri=<string>
   ,DayScheduleNameSat=<string>
<\ModifyDaySchedule>
```

24.2.125 Modify Destination

```
<ModifyDestination>
ModifyDestinationRequest(Please see WSDL
API:unsp__modifyDestination for more details)
ModifyDestination
```

```
   ,Name=<string>
   ,Signaling=<enum unsp__SignalingType>
   ,Bearer=<enum unsp__BearerCaps>
   ,DestinationAttributes=<enum
unsp__DestinationAttributes>
   ,Attributes=<enum unsp__Attributes>
   ,NumberPlanName=<string>
   ,PrefixAreaCode=<enum unsp__Attributes>
   ,PrefixCountryCode=<enum unsp__Attributes>
   ,ReRoute=<enum unsp__Attributes>
<\ModifyDestination>
```

24.2.126 Modify Endpoint

```
<ModifyEndpoint>
ModifyEndpointRequest(Please see WSDL
API:unsp__modifyEndpoint for more details)
ModifyEndpoint
   ,Name=<string>
   ,IpAddress=<string>
   ,AltIpAddress=<string>
   ,RasIpAddress=<string>
,ManagementIpAddress=<string>
   ,Port=<int>
   ,AltPort=<int>
   ,RasPort=<int>
   ,Registration=<short>
   ,TimeToLive=<double>
   ,AccountAuth=<enum unsp__AccountAuth>
   ,ProfileName=<string>
   ,Sessions=<long>
,MaxOriginatingSessionLimit=<long>
   ,MaxTerminatingSessionLimit=<long>
   ,Transport=<enum unsp__TransportType>
   ,Attributes=<long>
   ,AddressOfRecord=<string>
   ,ServerFailover=<enum unsp__EpFailover>
   ,KeysetUse=<enum unsp__KeysetUse>
   ,CountryCode=<string>
   ,AssociatedEndpointName=<string>
   ,IvtgNameDomain=<string>
   ,Attributes=<long>
,LooseRoutingURI=<string>
,EnablePrack=<bool>
,EndpointSrtpSupport=<enum unsp__SrtpSupport>
,EndpointLocationDomain=<string>
,SetNPIandTONtoUnknown=<bool>
,SendXferingNrIsoCgNrForXferedCalls=<bool>
,IncludeRestrNrsInFromHeader=<bool>
,Reserved2=<bool>
,Reserved3=<bool>
,Reserved4=<bool>
,EnableSessionTimer=<bool>
,UseEndpointDefaultHomeDnAsAuthNumber=<bool>
,UseSubscriberHomeDnAsAuthNumber=<bool>
,DisableLongCallAudit=<bool>
,SendReceiveImpactLevel=<bool>
,AddressOfRecord=<string>
```

```
,SupportCallbackPathReservation=<bool>
,SendProgStopCallProcTimer=<bool>
,VoiceMail=<bool>
,EnablePrack=<bool>
,SrtpSupport=<unsp__SrtpSupport
,AnatSupport=<enum unsp__Anat>
,LocationDomain=<string>
<\ModifyEndpoint>
```

24.2.127 Modify HomeDN

```
<ModifyHomeDn>
ModifyHomeDnRequest(Please see WSDL API:unsp__modifyHomeDn
for more details)
ModifyHomeDn
   ,ModifyType=<enum unsp__DnModifyType>
   ,StartCode=<string>
   ,EndCode=<string>
   ,OfficeCode=<string>
   ,DestinationName=<string>
   ,DestinationType=<enum unsp__DestinationType>
   ,SubscriberName=<string>
   ,NewDn=<string>
   ,EnableAssign=<enum unsp__CommandActivate>
   ,EnableMatch=<enum unsp__EnableMatch>
<\ModifyHomeDn>
```

24.2.128 Modify Intercept

```
<ModifyIntercept>
ModifyIntercept(Please see WSDL API:unsp__modifyIntercept
for more details)
ModifyIntercept
   ,Cycle=<long>
   ,Interval=<long>
   ,IsDefault=<enum unsp__IsDefault>
   ,DefaultName=<string>
   ,InterceptInfoName=<string>
<\ModifyIntercept>
```

24.2.129 Modify Media Gateway

```
<ModifyMediaGateway>
ModifyMediaGatewayRequest(Please see WSDL
API:unsp__modifyMediaGateway for more details)
ModifyMediaGateway
   ,AdminStatus=<enum unsp__ResourceStatus>
   ,GatewayName=<string>
   ,GatewayProtocolType=<enum unsp__GatewayProtocolType>
   ,GatewayProtocolVersion=<enum
unsp__GatewayProtocolVersion>
```

```
   ,MgCircuitFormat=<string>
   ,NetworkTypes=<int>
   ,CodecTypes=<int>
   ,MgAaddrProvision=<string>
   ,Timeout=<unsignedInt>
   ,RetryCount=<unsignedInt>
   ,IpAddrAllocMethod=<enum unsp__IPAddressAllocMethod>
   ,Fqdn=<string>
   ,MediaServerVersion=<enum unsp__MSVersion>
   ,TimerHist=<int>
   ,TimerMax=<int>
   ,TransmissionTimerLen=<int>
   ,FaxT38Relay=<enum unsp__T38FaxRelay>
   ,DtmfRelay=<enum unsp__DtmfRelay>
   ,MultiHomingFlag=<enum unsp__MultiHoming>
,OverloadSupport=<enum unsp__YesNo>
   ,OverloadAllowTwCalls=<enum unsp__YesNo>
   ,OverloadReturnCodeOne=<int>
   ,OverloadReturnCodeTwo=<int>
   ,OverloadReturnCodeThree=<int>
   ,OverloadGapIntervalOne=<int>
   ,OverloadGapIntervalTwo=<int>
   ,OverloadGapIntervalThree=<int>
<\ModifyMediaGateway>
```

24.2.130 Modify Office Code

```
<ModifyOfficeCode>
ModifyOfficeCodeRequest(Please see WSDL
API:unsp__modifyOfficeCode for more details)
ModifyOfficeCode
   ,OfficeCode=<string>
   ,CountryCodeLength=<short>
   ,AreaCodeLength=<short>
   ,Attributes=<enum unsp__Attributes>
   ,ChargeArea=<string>
<\ModifyOfficeCode>
```

24.2.131 Modify Origin Destination

```
<ModifyOriginDestination>
OriginDestinationRequest(Please see WSDL
API:unsp__modifyOriginDestination for more details)
ModifyOriginDestination
   ,Name=<string>
   ,Type=<enum unsp__OrigDestType>
<\ModifyOriginDestination>
```

24.2.132 Modify Origin Route

```
<ModifyOriginRoute>
```

```
ModifyOriginRouteRequest(Please see WSDL
API:unsp__modifyOriginRoute for more details)
ModifyOriginRoute
   ,OriginDestinationName=<string>
   ,RateAreaName=<string>
   ,DestinationName=<string>
   ,DestinationType=<enum unsp__OriginRouteDestinationType>
   ,IsupCircuitCode=<long>
<\ModifyOriginRoute>
```

24.2.133 Modify Period Schedule

```
<ModifyPeriodSchedule>
ModifyPeriodScheduleRequest(Please see WSDL
API:unsp__modifyPeriodSchedule for more details)
ModifyPeriodSchedule
   ,Name=<string>
   ,DayScheduleName=<string>
   ,TimeDestinationName=<string>
   ,StartTime=<string>
   ,StopTime=<string>
   ,DestinationName=<string>
<\ModifyPeriodSchedule>
```

24.2.134 Modify Route

```
<ModifyRoute>
ModifyRouteRequest(Please see WSDL API:unsp__modifyRoute
for more details)
ModifyRoute
   ,GenericRouteDestinationName=<string>
   ,GenericRouteDestinationSignaling=<enum
unsp__SignalingType>
   ,GenericRouteDestinationBearer=<enum unsp__BearerCaps>
   ,GenericRouteDestinationDestinationAttributes=<enum
unsp__DestinationAttributes>
   ,GenericRouteDestinationAttributes=<enum
unsp__Attributes>
   ,GenericRouteDestinationNumberPlanName=<string>
   ,GenericRouteDestinationPrefixAreaCode=<enum
unsp__Attributes>
   ,GenericRouteDestinationPrefixCountryCode=<enum
unsp__Attributes>
   ,GenericRouteDestinationReRoute=<enum unsp__Attributes>
   ,RouteId=<long>
   ,RouteName=<string>
   ,RouteReferenceName=<string>
   ,RouteReferenceType=<enum unsp__ReferenceType>
   ,RouteDigitsToDelete=<short>
   ,RouteDigitsToInsert=<string>
   ,RouteNatureOfAddress=<enum unsp__E164NatureOfAddr>
   ,RouteCacSignaling=<enum unsp__CacSignaling>
<\ModifyRoute>
```

24.2.135 Modify Treatment

```
<ModifyTreatment>
ModifyTreatmentData(Please see WSDL
API:unsp__modifyTreatment for more details)
ModifyTreatment
   ,Cycle=<long>
   ,Interval=<long>
   ,SipReasonId=<long>
   ,TreatmentInfoName=<string>
   ,ToneName=<string>
   ,ToneDuration=<long>
   ,ToneDestinationName=<string>
   ,ToneDestinationCircuit=<string>
   ,ToneDestinationType=<enum
unsp__InterceptDestinationType>
   ,AnnounceName=<string>
   ,AnnounceAnswer=<enum unsp__YesNo>
   ,AnnounceBargeIn=<enum unsp__YesNo>
   ,AnnounceDestinationName=<string>
   ,AnnounceDestinationCircuit=<string>
   ,AnnounceDestinationType=<enum
unsp__InterceptDestinationType>
<\ModifyTreatment>
```

24.2.136 Modify Trunk

```
<ModifyTrunk>
ModifyTrunkRequest(Please see WSDL API:unsp__modifyTrunk
for more details)
ModifyTrunk
   ,TrunkGroupName=<RtmTrunkGroupName>
   ,TrunkGroupNumber=<unsignedShort>
   ,TrunkNumberMin=<int>
   ,TrunkNumberMax=<int>
   ,AdminStatus=<enum unsp__ResourceStatus>
   ,HandlingGroupNumber=<short>
<\ModifyTrunk>
```

24.2.137 Modify Trunk Group

```
<ModifyTrunkGroup>
ModifyTrunkGroupRequest(Please see WSDL
API:unsp__modifyTrunkGroup for more details)
ModifyTrunkGroup
   ,TrunkGroupName=<RtmTrunkGroupName>
   ,TrunkGroupNumber=<unsignedInt>
   ,AdminStatus=<enum unsp__ResourceStatus>
,TrunkType=<enum unsp__TrunkType>
   ,TrunkGroupComment=<RtmLongString>
   ,SignalProfileName=<string>
   ,Usage=<enum unsp__TrunkGroupUsage>
   ,Owner=<RtmLongString>
   ,RemoteClli=<string>
   ,CDRRecordingIns=<enum unsp__CDRRecordingIns>
```

```
   ,SearchMethod=<enum unsp__SearchMethod>
   ,GainControl=<int>
   ,CotProcedure=<enum unsp__COTProcedure>
   ,CotFrequency=<int>
   ,CirCarrierIndicator=<enum
unsp__CircuitCarrierIndicator>
   ,AlarmCarrierIndicator=<enum
unsp__AlarmCarrierIndicator>
   ,HopCountInitValue=<int>
   ,CongestionControlAction=<enum
unsp__CongestionControlAction>
   ,CongestionControlLevel=<enum unsp__CongestionLevel>
   ,AccessTandem=<enum unsp__YesNo>
   ,LataDigits=<RtmLataDigit>
   ,DefaultDn=<RtmDefaultDn>
   ,RateAreaName=<RtmRateAreaName>
   ,ClassOfServiceName=<RtmClassOfServceName>
   ,NetworkIndicator=<enum unsp__NetworkIndicator>
   ,SigTranIndicator=<enum unsp__YesNo>
   ,PreferredTrunkUsage=<int>
   ,BothwayTrunksReserve=<int>
   ,PriorityTrunksReserve=<int>
   ,FarEndCarrierIdCode=<RtmFarEndCarrierIdCode>
   ,FarEndCarrierSwitchName=<RtmFarEndCarrierSwName>
   ,POIflag=<enum unsp__POIFlag>
   ,POIstage=<enum unsp__POIStage>
   ,POIChargeArea=<RtmPOIChargeArea>
   ,Language=<string>
<\ModifyTrunkGroup>
```

24.2.138 Prioritize Route

```
<PrioritizeRoute>
PrioritizeRouteRequest(Please see WSDL
API:unsp__prioritizeRoute for more details)
PrioritizeRoute
   ,Name=<string>
   ,DestinationName=<string>
   ,Signaling=<enum unsp__SignalingType>
   ,Bearer=<enum unsp__BearerCaps>
   ,RouteIndex=<long>
   ,ReferenceName=<string>
   ,ReferenceType=<enum unsp__ReferenceType>
   ,NewRouteIndex=<long>
<\PrioritizeRoute>
```

24.2.139 Prioritize Treatment

```
<PrioritizeTreatment>
PrioritizeTreatmentRequest(Please see WSDL
API:unsp__prioritizeTreatment for more details)
PrioritizeTreatment
   ,NewOrder=<enum unsp__TreatmentOrder>
   ,TreatmentInfoName=<string>
   ,InterceptInfoName=<string>
<\PrioritizeTreatment>
```

24.2.140 Remove Alias

```
<RemoveAlias>
RemoveAliasRequest(Please see WSDL API:unsp__removeAlias
for more details)
RemoveAlias
   ,NodeName=<string>
   ,NumberAliases=<short>
   ,AliasesNodeName=<string>
   ,AliasesAliasName=<string>
   ,AliasesAliasType=<enum unsp__AliasType>
   ,AliasesRegistration=<short>
<\RemoveAlias>
```

24.2.141 Remove Calling Location

```
<RemoveCallingLocation>
RemoveCallingLocationRequest(Please see WSDL
API:unsp__removeCallingLocation for more details)
RemoveCallingLocation
   ,Code=<string>
<\RemoveCallingLocation>
```

24.2.142 Remove Class Of Service

```
<RemoveClassService>
RemoveClassServiceRequest(Please see WSDL
API:unsp__removeClassService for more details)
RemoveClassService
   ,Name=<string>
   ,Number=<long>
   ,DefaultCpcCos=<long>
   ,DefaultCpcCosName=<string>
<\RemoveClassService>
```

24.2.143 Remove Code Process

```
<RemoveCodeProcess>
RemoveCodeProcessRequest(Please see WSDL
API:unsp__removeCodeProcess for more details)
RemoveCodeProcess
   ,CodeProcessName=<string>
<\RemoveCodeProcess>
```

24.2.144 Remove Day Schedule

```
<RemoveDaySchedule>
RemoveDayScheduleRequest(Please see WSDL
API:unsp__removeDaySchedule for more details)
RemoveDaySchedule
   ,Name=<string>
   ,TimeDestinationName=<string>
<\RemoveDaySchedule>
```

24.2.145 Remove Destination

```
<RemoveDestination>
RemoveDestinationRequest(Please see WSDL
API:unsp__removeDestination for more details)
RemoveDestination
   ,Name=<string>
   ,Signaling=<enum unsp__SignalingType>
   ,Bearer=<enum unsp__BearerCaps>
   ,DestinationAttributes=<enum
unsp__DestinationAttributes>
   ,Attributes=<enum unsp__Attributes>
   ,NumberPlanName=<string>
<\RemoveDestination>
```

24.2.146 Remove Endpoint

```
<RemoveEndpoint>
RemoveEndpointRequest(Please see WSDL
API:unsp__removeEndpoint for more details)
RemoveEndpoint
   ,EndpointType=<enum unsp__EndpointType>
   ,NumberEndpoints=<short>
   ,EndpointsName=<string>
<\RemoveEndpoint>
```

24.2.147 Remove HomeDN

```
<RemoveHomeDn>
RemoveHomeDnRequest(Please see WSDL API:unsp__removeHomeDn
for more details)
RemoveHomeDn
   ,StartCode=<string>
   ,EndCode=<string>
   ,OfficeCode=<string>
<\RemoveHomeDn>
```

24.2.148 Remove Intercept

```
<RemoveIntercept>
RemoveInterceptRequest(Please see WSDL
API:unsp__removeIntercept for more details)
RemoveIntercept
   ,RemoveName=<boolean>
   ,InterceptInfoName=<string>
<\RemoveIntercept>
```

24.2.149 Remove Media Gateway Circuits

```
<RemoveMediaGatewayCircuits>
RemoveMediaGatewayCircuitsRequest(Please see WSDL
API:unsp__removeMediaGatewayCircuits for more details)
RemoveMediaGatewayCircuits
   ,GatewayName=<string>
   ,TrunkGroupInfo=<enum unsp__TrunkGroupNo>
   ,StartCircuitId=<string>
   ,EndCircuitId=<string>
<\RemoveMediaGatewayCircuits>
```

24.2.150 Remove Media Gateway

```
<RemoveMediaGateway>
RemoveMediaGatewayRequest(Please see WSDL
API:unsp__removeMediaGateway for more details)
RemoveMediaGateway
   ,GatewayName=<string>
<\RemoveMediaGateway>
```

24.2.151 Remove Office Code

```
<RemoveOfficeCode>
RemoveOfficeCodeRequest(Please see WSDL
API:unsp__removeOfficeCode for more details)
RemoveOfficeCode
   ,OfficeCode=<string>
<\RemoveOfficeCode>
```

24.2.152 Remove Origin Destination

```
<RemoveOriginDestination>
OriginDestinationRequest(Please see WSDL
API:unsp__removeOriginDestination for more details)
RemoveOriginDestination
```

```
   ,Name=<string>
<\RemoveOriginDestination>
```

24.2.153 Remove Origin Route

```
<RemoveOriginRoute>
RemoveOriginRouteRequest(Please see WSDL
API:unsp__removeOriginRoute for more details)
RemoveOriginRoute
   ,OriginDestinationName=<string>
   ,RateAreaName=<string>
<\RemoveOriginRoute>
```

24.2.154 Remove Period Schedule

```
<RemovePeriodSchedule>
RemovePeriodScheduleRequest(Please see WSDL
API:unsp__removePeriodSchedule for more details)
RemovePeriodSchedule
   ,Name=<string>
   ,DayScheduleName=<string>
   ,TimeDestinationName=<string>
   ,StartTime=<string>
<\RemovePeriodSchedule>
```

24.2.155 Remove Rate Area

```
<RemoveRateArea>
RemoveRateAreaRequest(Please see WSDL
API:unsp__removeRateArea for more details)
RemoveRateArea
   ,Group=<long>
   ,Name=<string>
   ,NpaNxx=<string>
   ,SlhrName=<string>
   ,PointCode=<long>
   ,NodeVariant=<enum unsp__NodeVariant>
   ,Lata=<string>
<\RemoveRateArea>
```

24.2.156 Remove Time Destination

```
<RemoveTimeDestination>
RemoveTimeDestination(Please see WSDL
API:unsp__removeTimeDestination for more details)
RemoveTimeDestination
   ,Name=<string>
```

```
<\RemoveTimeDestination>
```

24.2.157 Remove Trunk

```
<RemoveTrunk>
RemoveTrunkRequest(Please see WSDL API:unsp__removeTrunk
for more details)
RemoveTrunk
   ,TrunkGroupName=<RtmTrunkGroupName>
   ,TrunkGroupNumber=<unsignedShort>
   ,TrunkNumberMin=<int>
   ,TrunkNumberMax=<int>
<\RemoveTrunk>
```

24.2.158 Remove Trunk Group

```
<RemoveTrunkGroup>
RemoveTrunkGroupRequest(Please see WSDL
API:unsp__removeTrunkGroup for more details)
RemoveTrunkGroup
   ,TrunkGroupName=<RtmTrunkGroupName>
   ,TrunkGroupNumber=<unsignedInt>
<\RemoveTrunkGroup>
```

24.2.159 Unassign Day Schedule

```
<UnassignDaySchedule>
UnassignDayScheduleRequest(Please see WSDL
API:unsp__unassignDaySchedule for more details)
UnassignDaySchedule
   ,TimeDestinationName=<string>
   ,UnassignSun=<boolean>
   ,UnassignMon=<boolean>
   ,UnassignTue=<boolean>
   ,UnassignWed=<boolean>
   ,UnassignThu=<boolean>
   ,UnassignFri=<boolean>
   ,UnassignSat=<boolean>
<\UnassignDaySchedule>
```

24.2.160 Unassign Route

```
<UnassignRoute>
UnassignRouteRequest(Please see WSDL
API:unsp__unassignRoute for more details)
UnassignRoute
   ,RouteId=<long>
```

```
   ,NumRoutes=<long>
   ,GenericRouteDestinationName=<string>
   ,GenericRouteDestinationSignaling=<enum
unsp__SignalingType>
   ,GenericRouteDestinationBearer=<enum unsp__BearerCaps>
   ,GenericRouteDestinationDestinationAttributes=<enum
unsp__DestinationAttributes>
   ,GenericRouteDestinationAttributes=<enum
unsp__Attributes>
   ,GenericRouteDestinationNumberPlanName=<string>
   ,GenericRouteDestinationPrefixAreaCode=<enum
unsp__Attributes>
   ,GenericRouteDestinationPrefixCountryCode=<enum
unsp__Attributes>
   ,GenericRouteDestinationReRoute=<enum unsp__Attributes>
   ,RouteListId=<long>
   ,RouteListName=<string>
<\UnassignRoute>
```

24.2.161 Unassign Treatment

```
<UnassignTreatment>
UnassignTreatmentId(Please see WSDL
API:unsp__unassignTreatment for more details)
UnassignTreatment
   ,Name=<string>
   ,Order=<enum unsp__TreatmentOrder>
   ,InterceptInfoName=<string>
<\UnassignTreatment>
```

24.2.162 Create Enum Server

```
<EnumServer>
CreateEnumServerRequest(Please see WSDL
API:unsp__createEnumServer for more details)
EnumServer
   ,EnumServerName=<string>
   ,PrimaryIpAddress=<string>
   ,SecondaryIpAddress=<string>
   ,CacheSize=<short>
<\EnumServer>
```

24.2.163 Modify Enum Server

```
<ModifyEnumServer>
ModifyEnumServerRequest(Please see WSDL
API:unsp__modifyEnumServer for more details)
ModifyEnumServer
   ,EnumServerName=<string>
   ,PrimaryIpAddress=<string>
```

```
   ,SecondaryIpAddress=<string>
   ,CacheSize=<short>
<\ModifyEnumServer>
```

24.2.164 Remove Enum Server

```
<RemoveEnumServer>
RemoveEnumServerRequest(Please see WSDL
API:unsp__removeEnumServer for more details)
RemoveEnumServer
   ,EnumServerName=<string>
<\RemoveEnumServer>
```

24.2.165 Create Operator Enum

```
<OperatorEnum>
CreateOperatorEnumRequest(Please see WSDL
API:unsp__createOperatorEnum for more details)
OperatorEnum
   ,OperatorEnumName=<string>
   ,OperatorEnumTier0ZoneName=<string>
   ,EnumServerName=<string>
<\OperatorEnum>
```

24.2.166 Modify Operator Enum

```
<ModifyOperatorEnum>
ModifyOperatorEnumRequest(Please see WSDL
API:unsp__modifyOperatorEnum for more details)
ModifyOperatorEnum
   ,OperatorEnumName=<string>
   ,OperatorEnumTier0ZoneName=<string>
   ,EnumServerName=<string>
   ,OperatorEnumUpdateFlags=<int>
<\ModifyOperatorEnum>
```

24.2.167 Remove Operator Enum

```
<RemoveOperatorEnum>
RemoveOperatorEnumRequest(Please see WSDL
API:unsp__removeOperatorEnum for more details)
RemoveOperatorEnum
   ,OperatorEnumName=<string>
<\RemoveOperatorEnum>
```

24.2.168 Set RTP Parameters

```
<RTPSET>
RTPSET - " Set the RTP parameters" (Please see WSDL
API:unsp__SetConfigParmData in Provisioning Doc for
Attribute details)
,RTPNAME=<string>
,RTPVALUE=<string>
</RTPSET>
```

24.2.169 Create Disconnect DN

This operation is specifically to allow the export of subscribers that have been previously disconnected so that they can be imported in the disconnected state into a new switch if necessary.

```
<CREATE_DISCONNECT_DN>
SERVICEID:
Type:string
Length:(1-15) Default:-
Mandatory:yes
Comment:"The service Id of the subscriber"
INTERCEPTANNOUNCEMENT:
Type:enum
Values:"[None Disconnect NumberChange]" Default:-
Mandatory:yes
Comment:"Default Intercept for the subscriber"
Type:enum
NEWSERVICEID:
Type:string
Length:(1-15) Default:""
Mandatory:no
Comment:"Used when a subscriber's ServiceId has changed"
</CREATE_DISCONNECT_DN>
```

24.2.170 Update Dn Reservation

```
<BGRESERVATION>
BGRESERVATION - "Get reserved dn's for a bg" (Please see
WSDL API:unsp__UpdateDnReservation in Provisioning Doc for
Attribute details)
,BGNAME=<string>
,BGDNOFFICECODE=<string>
,BGDNSTARTCODE=<string>
,BGDNENDCODE=<string>
</BGRESERVATION>
```

24.2.171 Create a Parking Lot

```
<PARKINGLOTCREATE>
PARKINGLOTCREATE - Creates a Parking Lot (Please see WSDL
API: unsp__CreateParkingLotRequest  )
   , BGNAME,=<string>
   , PARKINGLOTNAME=<string>
   , RECALLTIMER=<unsigned int>
   , PARKINGSTART=<unsigned long int>
   , PARKINGEND=<unsigned long int>
</PARKINLOTCREATE>
```

24.2.172 Update a Parking Lot

```
<PARKINGLOTUPDATE>
PARKINGLOTUPDATE - Update a Parking Lot (Please see WSDL
API: unsp__UpdateParkingLotRequest  )
   , BGNAME,=<string>
   , PARKINGLOTNAME=<string>
   , NEWPARKINGLOTNAME=<string>
   , RECALLTIMER=<unsigned int>
   , PARKINGSTART=<unsigned long int>
   , PARKINGEND=<unsigned long int>
</PARKINLOTUPDATE>
```

24.2.173 Delete a Parking Lot

```
<PARKINGLOTDELETE>
PARKINGLOTDELETE - Delete a Parking Lot (Please see WSDL
API: unsp__DeleteParkingLotRequest  )
   , BGNAME,=<string>
   , PARKINGLOTNAME=<string>
</PARKINLOTDELETE>
```

24.2.174 Add Community Group

```
<ACG>
ACG - Adds a Community Group" (Please see WSDL
API:unsp__AddCommunityGroup in Provisioning Doc for
Attribute details)
, SUBSCRIBERDN,=<string>
, PERIODICTONE=<string>
</ACG>
```

24.2.175 Add Community Group Member

```
<ACGM>
ACGM - Adds a Community Group Member" (Please see WSDL
API:unsp__AddCommunityGroupMember in Provisioning Doc for
Attribute details)
```

```
,SUBSCRIBERDN,=<string>
,COMNUMBER=<int>
,DESTNUMBER=<string>
,DESTNAME=<string>
,INCBLOCK=<bool>
</ACGM>
```

24.2.176 Modify Community Group

```
<MCG>
MCG - Modifies a Community Group" (Please see WSDL
API:unsp__AddCommunityGroup in Provisioning Doc for
Attribute details)
, SUBSCRIBERDN,=<string>
, PERIODICTONE=<string>
</MCG>
```

24.2.177 Modify Community Group Member

```
<MCGM>
MCGM - Updates a Community Group Member" (Please see WSDL
API:unsp__ModifyCommunityGroupMember in Provisioning Doc
for Attribute details)
,SUBSCRIBERDN=<string>
,COMNUMBER=<int>
,NEWCOMNUMBER=<int>
,DESTNUMBER=<string>
,DESTNAME=<string>
,INCBLOCK=<bool>
</MCGM>
```

24.2.178 Remove Community Group Member

```
<RCGM>
RCGM - "Removes a Community Group Member" (Please see WSDL
API:unsp__RemoveCommunityGroupMember in Provisioning Doc
for Attribute details)
,SUBSCRIBERDN=<string>
,COMNUMBER=<int>
</RCGM>
```

24.2.179 Remove Community Group

```
<RCG>
RCG - "Removes a Community Group" (Please see WSDL
API:unsp__RemoveCommunityGroup in Provisioning Doc for
Attribute details)
,SUBSCRIBERDN=<string>
</RCG>
```

24.2.180 Update Subscriber Mobile Info

```
<UMI>
UMI - Updates Subscriber Mobile Info(Please see WSDL
API:unsp__UpdateMobileInfo
,SUBSCRIBERDN=<string>
,MOBILEFLAG=<bool>
,MAINDEVICEDN=<string>
,CELLDEVICENR=<string>
,MOBRINGDUR=<int>
,MAINRINGDUR=<int>
,CELLRINGDUR=<int>
</UMI>
```

24.2.181 Create Traffic Type

```
<TRAFFICTYPECREATE>
TRAFFICTYPECREATE - Create TrafficType (Please see wsdl API:
unsp__CreateTrafficType)
,TTNAME=<string>
</TRAFFICTYPECREATE>
```

24.2.182 Modify Traffic Type

```
<TRAFFICTYPEMODIFY>
TRAFFICTYPEMODIFY - Modify TrafficType (Please see wsdl API:
unsp__ModifyTrafficType
,TTNAME=<string>
,TTNEWNAME=<string>
</TRAFFICTYPEMODIFY>
```

24.2.183 Delete Traffic Type

```
<TRAFFICTYPEDELETE>
TRAFFICTYPEDELETE - Delete TrafficType (Please see wsdl API:
unsp__DeleteTrafficType
,TTNAME=<string>
,TTNEWNAME=<string>
</TRAFFICTYPEDELETE>
```

24.2.184 Create Class Of Restrictions

```
<CORCREATE>
CORCREATE - Create Class of restrictions (Please see wsdl
API: unsp__CreateCOR
CORNAME=<string>
TTNAME_0=<string>
TTNAME_1=<string>
TTNAME_2=<string>
TTNAME_3=<string>
```

```
TTNAME_4=<string>
TTNAME_5=<string>
TTNAME_6=<string>
TTNAME_7=<string>
TTNAME_8=<string>
TTNAME_9=<string>
TTNAME_10=<string>
TTNAME_11=<string>
TTNAME_12=<string>
TTNAME_13=<string>
TTNAME_14=<string>
TTNAME_15=<string>
TTNAME_16=<string>
TTNAME_17=<string>
TTNAME_18=<string>
TTNAME_19=<string>
TTNAME_20=<string>
TTNAME_21=<string>
TTNAME_22=<string>
TTNAME_23=<string>
TTNAME_24=<string>
TTNAME_25=<string>
TTNAME_26=<string>
TTNAME_27=<string>
TTNAME_28=<string>
TTNAME_29=<string>
TTNAME_30=<string>
TTNAME_31=<string>
TTNAME_32=<string>
TTNAME_33=<string>
TTNAME_34=<string>
TTNAME_35=<string>
TTNAME_36=<string>
TTNAME_37=<string>
TTNAME_38=<string>
TTNAME_39=<string>
TTNAME_40=<string>
TTNAME_41=<string>
TTNAME_42=<string>
TTNAME_43=<string>
TTNAME_44=<string>
TTNAME_45=<string>
TTNAME_46=<string>
TTNAME_47=<string>
TTNAME_48=<string>
TTNAME_49=<string>
TTNAME_50=<string>
TTNAME_51=<string>
TTNAME_52=<string>
TTNAME_53=<string>
TTNAME_54=<string>
TTNAME_55=<string>
TTNAME_56=<string>
TTNAME_57=<string>
TTNAME_58=<string>
TTNAME_59=<string>
TTNAME_60=<string>
TTNAME_61=<string>
TTNAME_62=<string>
TTNAME_63=<string>
TTNAME_64=<string>
TTNAME_65=<string>
```

```
TTNAME_66=<string>
TTNAME_67=<string>
TTNAME_68=<string>
TTNAME_69=<string>
TTNAME_70=<string>
TTNAME_71=<string>
TTNAME_72=<string>
TTNAME_73=<string>
TTNAME_74=<string>
TTNAME_75=<string>
TTNAME_76=<string>
TTNAME_77=<string>
TTNAME_78=<string>
TTNAME_79=<string>
TTNAME_80=<string>
TTNAME_81=<string>
TTNAME_82=<string>
TTNAME_83=<string>
TTNAME_84=<string>
TTNAME_85=<string>
TTNAME_86=<string>
TTNAME_87=<string>
TTNAME_88=<string>
TTNAME_89=<string>
TTNAME_90=<string>
TTNAME_91=<string>
TTNAME_92=<string>
TTNAME_93=<string>
TTNAME_94=<string>
TTNAME_95=<string>
TTNAME_96=<string>
TTNAME_97=<string>
TTNAME_98=<string>
TTNAME_99=<string>
TTNAME_100=<string>
TTNAME_101=<string>
TTNAME_102=<string>
TTNAME_103=<string>
TTNAME_104=<string>
TTNAME_105=<string>
TTNAME_106=<string>
TTNAME_107=<string>
TTNAME_108=<string>
TTNAME_109=<string>
TTNAME_110=<string>
TTNAME_111=<string>
TTNAME_112=<string>
TTNAME_113=<string>
TTNAME_114=<string>
TTNAME_115=<string>
TTNAME_116=<string>
TTNAME_117=<string>
TTNAME_118=<string>
TTNAME_119=<string>
TTNAME_120=<string>
TTNAME_121=<string>
TTNAME_122=<string>
TTNAME_123=<string>
TTNAME_124=<string>
TTNAME_125=<string>
TTNAME_126=<string>
</CORCREATE>
```

24.2.185 Modify Class Of Restrictions

```
<<CORMODIFY>
CORMODIFY - Modify Class of restrictions (Please see wsdl
API: unsp__ModifyCOR
CORNAME=<string>
TTNAME_0=<string>
TTNAME_1=<string>
TTNAME_2=<string>
TTNAME_3=<string>
TTNAME_4=<string>
TTNAME_5=<string>
TTNAME_6=<string>
TTNAME_7=<string>
TTNAME_8=<string>
TTNAME_9=<string>
TTNAME_10=<string>
TTNAME_11=<string>
TTNAME_12=<string>
TTNAME_13=<string>
TTNAME_14=<string>
TTNAME_15=<string>
TTNAME_16=<string>
TTNAME_17=<string>
TTNAME_18=<string>
TTNAME_19=<string>
TTNAME_20=<string>
TTNAME_21=<string>
TTNAME_22=<string>
TTNAME_23=<string>
TTNAME_24=<string>
TTNAME_25=<string>
TTNAME_26=<string>
TTNAME_27=<string>
TTNAME_28=<string>
TTNAME_29=<string>
TTNAME_30=<string>
TTNAME_31=<string>
TTNAME_32=<string>
TTNAME_33=<string>
TTNAME_34=<string>
TTNAME_35=<string>
TTNAME_36=<string>
TTNAME_37=<string>
TTNAME_38=<string>
TTNAME_39=<string>
TTNAME_40=<string>
TTNAME_41=<string>
TTNAME_42=<string>
TTNAME_43=<string>
TTNAME_44=<string>
TTNAME_45=<string>
TTNAME_46=<string>
TTNAME_47=<string>
TTNAME_48=<string>
TTNAME_49=<string>
TTNAME_50=<string>
TTNAME_51=<string>
TTNAME_52=<string>
TTNAME_53=<string>
TTNAME_54=<string>
```

```
TTNAME_55=<string>
TTNAME_56=<string>
TTNAME_57=<string>
TTNAME_58=<string>
TTNAME_59=<string>
TTNAME_60=<string>
TTNAME_61=<string>
TTNAME_62=<string>
TTNAME_63=<string>
TTNAME_64=<string>
TTNAME_65=<string>
TTNAME_66=<string>
TTNAME_67=<string>
TTNAME_68=<string>
TTNAME_69=<string>
TTNAME_70=<string>
TTNAME_71=<string>
TTNAME_72=<string>
TTNAME_73=<string>
TTNAME_74=<string>
TTNAME_75=<string>
TTNAME_76=<string>
TTNAME_77=<string>
TTNAME_78=<string>
TTNAME_79=<string>
TTNAME_80=<string>
TTNAME_81=<string>
TTNAME_82=<string>
TTNAME_83=<string>
TTNAME_84=<string>
TTNAME_85=<string>
TTNAME_86=<string>
TTNAME_87=<string>
TTNAME_88=<string>
TTNAME_89=<string>
TTNAME_90=<string>
TTNAME_91=<string>
TTNAME_92=<string>
TTNAME_93=<string>
TTNAME_94=<string>
TTNAME_95=<string>
TTNAME_96=<string>
TTNAME_97=<string>
TTNAME_98=<string>
TTNAME_99=<string>
TTNAME_100=<string>
TTNAME_101=<string>
TTNAME_102=<string>
TTNAME_103=<string>
TTNAME_104=<string>
TTNAME_105=<string>
TTNAME_106=<string>
TTNAME_107=<string>
TTNAME_108=<string>
TTNAME_109=<string>
TTNAME_110=<string>
TTNAME_111=<string>
TTNAME_112=<string>
TTNAME_113=<string>
TTNAME_114=<string>
TTNAME_115=<string>
TTNAME_116=<string>
```

```
TTNAME_117=<string>
TTNAME_118=<string>
TTNAME_119=<string>
TTNAME_120=<string>
TTNAME_121=<string>
TTNAME_122=<string>
TTNAME_123=<string>
TTNAME_124=<string>
TTNAME_125=<string>
TTNAME_126=<string>
</CORMODIFY>
```

24.2.186 Create Call Pickup Group

```
<ACPU>
ACPU - "Adds Call Pickup Group" (Please see WSDL
API:unsp__AddCpuGroup in Provisioning Doc for Attribute
details)
,BGNAME=<string>
,CPUID=<int>
,CPUNAME=<wstring>
,NOTIFICATIONDELAYTIMER=<int>
,CIDPRESENTATION=<enum>
</ACPU>
```

24.2.187 Modify Call Pickup Group

```
<MCPU>
MCPU - "Modifies a Call Pickup Group" (Please see WSDL
API:unsp__ModifyCpuGroup in Provisioning Doc for Attribute
details)
,BGNAME=<string>
,CPUID=<int>
,CPUNAME=<wstring>
,NOTIFICATIONDELAYTIMER=<int>
,CIDPRESENTATION=<enum>
</MCPU>
```

24.2.188 Delete Call Pickup Group

```
<RCPU>
RCPU - "Deletes a Call Pickup Group" (Please see WSDL
API:unsp__DeleteBGCpu2 in Provisioning Doc for Attribute
details)
,BGNAME=<string>
,CPUID=<int>
</RCPU>
```

24.2.189 Add Local Call Pickup Group member

```
<ALOCALCPU>
```

```
ALOCALCPU - "Adds Member to a Call Pickup Group" (Please see
WSDL API:unsp__AddLocalCpuMembers in Provisioning Doc for
Attribute details)
,BGNAME=<string>
,CPUID=<int>
,LOCALCPU_SERVICEID=<string>
,LOCALCPU_CIDCALLSTOMEMBER=<enum>
,LOCALCPU_CIDNOTIFICATIONTOMEMBER=<enum>
</ALOCALCPU>
```

24.2.190 Modify Local Call Pickup Group member

```
<MLOCALCPU>
MLOCALCPU - "Modifies a Member of a Call Pickup Group"
(Please see WSDL API:unsp__ModifyLocalCpuMembers in
Provisioning Doc for Attribute details)
,BGNAME=<string>
,CPUID=<int>
,LOCALCPU_SERVICEID=<string>
,LOCALCPU_CIDCALLSTOMEMBER=<enum>
,LOCALCPU_CIDNOTIFICATIONTOMEMBER=<enum>
</MLOCALCPU>
```

24.2.191 Remove local Call Pickup Group member

```
<RLOCALCPU>
RLOCALCPU - "Removes a Member of a Call Pickup Group"
(Please see WSDL API:unsp__RemoveLocalCpuMembers in
Provisioning Doc for Attribute details)
,BGNAME=<string>
,CPUID=<int>
,LOCALCPU_SERVICEID=<string>
</RLOCALCPU>
```

24.2.192 Add Remote Call Pickup Group

```
<AREMOTECPU>
AREMOTECPU - "Adds a remote member Member to a Call Pickup
Group" (Please see WSDL API:unsp__AddRemoteCpuGroups in
Provisioning Doc for Attribute details)
,BGNAME=<string>
,CPUID=<int>
,REMOTECPUGROUPNUMBER=<int>
,REMOTECPUGROUPACCESSCODE=<string>
,REMOTECPUGROUPDESCRIPTOR=<string>
,REMOTECPUGROUPHIPATHDX=<bool>
</AREMOTECPU>
```

24.2.193 Modify Remote Call Pickup Group

```
<MREMOTECPU>
```

```
MREMOTECPU - "Modify a remote member Member of a Call Pickup
Group" (Please see WSDL API:unsp__ModifyRemoteCpuGroups in
Provisioning Doc for Attribute details)
,BGNAME=<string>
,CPUID=<int>
,REMOTECPUGROUPNUMBER=<int>
,REMOTECPUGROUPACCESSCODE=<string>
,REMOTECPUGROUPDESCRIPTOR=<string>
,REMOTECPUGROUPHIPATHDX=<bool>
</MREMOTECPU>
```

24.2.194 Remove Remote Call Pickup Group

```
<RREMOTECPU>
RREMOTECPU - "Remove a remote member Member of a Call Pickup
Group" (Please see WSDL API:unsp__RemoveRemoteCpuGroups in
Provisioning Doc for Attribute details)
,BGNAME=<string>
,CPUID=<int>
,REMOTECPUGROUPNUMBER=<int>
,REMOTECPUGROUPACCESSCODE=<string>
</RREMOTECPU>
```

24.2.195 Create Branch Office

```
<CBO>
CBO - "Creates a Branch Office" (Please
see WSDL API:unsp__createBranchOffice in
Provisioning Doc for Attribute details)
,BRANCHOFFICENAME=<string>
,BGNAME=<string>
,REPRESENTATIVEENDPOINTNAME=<string>
,DIDPOOLUPDATE=<enum>
,BOOFFICECODE=<string>
,RATEAREA=<string>
,NUMBERPLANNAME=<string>
,BOATTRIBUTES=<bool>
,USERNAME=<string>
,PASSWORD=<string>
,NUMDIDDNCODES=<int>
,DIDDNCODE0=<string>
,DIDDNCODE1=<string>
,DIDDNCODE2=<string>
,DIDDNCODE3=<string>
,DIDDNCODE4=<string>
,DIDDNCODE5=<string>
,DIDDNCODE6=<string>
,DIDDNCODE7=<string>
,DIDDNCODE8=<string>
,DIDDNCODE9=<string>
,DIDDNCODE10=<string>
,DIDDNCODE11=<string>
,DIDDNCODE12=<string>
,DIDDNCODE13=<string>
,DIDDNCODE14=<string>
,DIDDNCODE15=<string>
```

```
,DIDDNCODE16=<string>
,DIDDNCODE17=<string>
,DIDDNCODE18=<string>
,DIDDNCODE19=<string>
,DIDDNCODE20=<string>
,DIDDNCODE21=<string>
,DIDDNCODE22=<string>
,DIDDNCODE23=<string>
,DIDDNCODE24=<string>
,DIDDNCODE25=<string>
,DIDDNCODE26=<string>
,DIDDNCODE27=<string>
,DIDDNCODE28=<string>
,DIDDNCODE29=<string>
,DIDDNCODE30=<string>
,DIDDNCODE31=<string>
,DIDDNCODE32=<string>
,DIDDNCODE33=<string>
,DIDDNCODE34=<string>
,DIDDNCODE35=<string>
,DIDDNCODE36=<string>
,DIDDNCODE37=<string>
,DIDDNCODE38=<string>
,DIDDNCODE39=<string>
,DIDDNCODE40=<string>
,DIDDNCODE41=<string>
,DIDDNCODE42=<string>
,DIDDNCODE43=<string>
,DIDDNCODE44=<string>
,DIDDNCODE45=<string>
,DIDDNCODE46=<string>
,DIDDNCODE47=<string>
,DIDDNCODE48=<string>
,DIDDNCODE49=<string>
,DIDDNCODE50=<string>
,DIDDNCODE51=<string>
,DIDDNCODE52=<string>
,DIDDNCODE53=<string>
,DIDDNCODE54=<string>
,DIDDNCODE55=<string>
,DIDDNCODE56=<string>
,DIDDNCODE57=<string>
,DIDDNCODE58=<string>
,DIDDNCODE59=<string>
,DIDDNCODE60=<string>
,DIDDNCODE61=<string>
,DIDDNCODE62=<string>
,DIDDNCODE63=<string>
</CBO>
```

24.2.196 Modify Branch Office

```
<MBO>
MBO - "Modifies a Branch Office" (Please see WSDL
API:unsp__modifyBranchOffice in Provisioning Doc for
Attribute details)
,BRANCHOFFICENAME=<string>
,NEWBRANCHOFFICENAME=<string>
,BGNAME=<string>
```

```
,REPRESENTATIVEENDPOINTNAME=<string>
,DIDPOOLUPDATE=<enum>
,BOOFFICECODE=<string>
,RATEAREA=<string>
,NUMBERPLANNAME=<string>
,BOATTRIBUTES=<bool>
,USERNAME=<string>
,PASSWORD=<string>
,NUMDIDDNCODES=<int>
,DIDDNCODE0=<string>
,DIDDNCODE1=<string>
,DIDDNCODE2=<string>
,DIDDNCODE3=<string>
,DIDDNCODE4=<string>
,DIDDNCODE5=<string>
,DIDDNCODE6=<string>
,DIDDNCODE7=<string>
,DIDDNCODE8=<string>
,DIDDNCODE9=<string>
,DIDDNCODE10=<string>
,DIDDNCODE11=<string>
,DIDDNCODE12=<string>
,DIDDNCODE13=<string>
,DIDDNCODE14=<string>
,DIDDNCODE15=<string>
,DIDDNCODE16=<string>
,DIDDNCODE17=<string>
,DIDDNCODE18=<string>
,DIDDNCODE19=<string>
,DIDDNCODE20=<string>
,DIDDNCODE21=<string>
,DIDDNCODE22=<string>
,DIDDNCODE23=<string>
,DIDDNCODE24=<string>
,DIDDNCODE25=<string>
,DIDDNCODE26=<string>
,DIDDNCODE27=<string>
,DIDDNCODE28=<string>
,DIDDNCODE29=<string>
,DIDDNCODE30=<string>
,DIDDNCODE31=<string>
,DIDDNCODE32=<string>
,DIDDNCODE33=<string>
,DIDDNCODE34=<string>
,DIDDNCODE35=<string>
,DIDDNCODE36=<string>
,DIDDNCODE37=<string>
,DIDDNCODE38=<string>
,DIDDNCODE39=<string>
,DIDDNCODE40=<string>
,DIDDNCODE41=<string>
,DIDDNCODE42=<string>
,DIDDNCODE43=<string>
,DIDDNCODE44=<string>
,DIDDNCODE45=<string>
,DIDDNCODE46=<string>
,DIDDNCODE47=<string>
,DIDDNCODE48=<string>
,DIDDNCODE49=<string>
,DIDDNCODE50=<string>
,DIDDNCODE51=<string>
,DIDDNCODE52=<string>
```

```
,DIDDNCODE53=<string>
,DIDDNCODE54=<string>
,DIDDNCODE55=<string>
,DIDDNCODE56=<string>
,DIDDNCODE57=<string>
,DIDDNCODE58=<string>
,DIDDNCODE59=<string>
,DIDDNCODE60=<string>
,DIDDNCODE61=<string>
,DIDDNCODE62=<string>
,DIDDNCODE63=<string>
</MBO
```

24.2.197 Remove Branch Office

```
<RBO>
RBO - "Removes a Branch Office" (Please
see WSDL API:unsp__removeBranchOffice in
Provisioning Doc for Attribute details)
,BRANCHOFFICENAME=<string>
,BGNAME=<string>
</RBO>
```

25 Subscriber Provisioning Result Information

This section describes the information that is returned as a result of an operation being performed.

25.1 Result Code Information

All error codes are eight digits long. The error codes will have the following format:

```
subRC_<category>_<object>_<attribute>_<error>.
```

The most significant digit (the first digit) is the category. The next two digits identify the object. The fourth and fifth digits are the sub-object. Digits six to eight represent the specific error. Sub-objects are defined only for objects EP, BG, and FEAT. With this numbering scheme, we support 10 categories, 99 objects, 99 sub-objects for each object, and 999 unique return codes for each object/subobject.

The category is at a higher level and can be one of those given in Table 944.

Code	Name	Description
1	DB	Database errors (get, insert, delete). In some cases, the request may be successful if retried after 30 seconds.
2	SYS	System errors (memory errors, file access)
3	APP	An error detected in the client request or the request is invalid for the current state of the system objects.
4	WAR	Warnings indicating the operation may be completed with errors
9	INT	Internal errors. In some cases, the objects being accessed may have been left in a corrupted state.

Table 944 Error Code Categories

Note that for errors in category 1 (DB) or 9 (INT), the SOAP Server process may restart in an attempt to clear up any potential data corruption. In this case, the SOAP Server will not be available to handle new requests for approximately three seconds following the error.

The object represents the resource being manipulated. Can be one of those given in Table 945.

Code	Name	Description
1	SUB	Subscriber errors
2	EP	Endpoint errors
3	BG	Business group
4	PNP	Private Number Plan

Table 945 Error Code Objects (Sheet 1 of 2)

Code	Name	Description
5	FP	Feature Profile
6	MLHG	Hunt Group
7	QOS	Quality of Service
9	JOB	Job errors
10	UTIL	Utilities (language, PIC, …)
11	FEAT	Feature
12	COMM	Error in common objects

Table 945 Error Code Objects (Sheet 2 of 2)

The subobjects within a Business Group can those given in Table 946.

Code	Name	Description
1		Common BG errors
2	AUTH	Authorization code errors
3	DEPT	Department errors
4	DP	Dialing Plan errors
5	FEAT	Feature errors
6	MAIN_NUM	Main Number errors
7	NUMBER_PLAN	Number Plan errors
8	SUBNET	Subnet errors

Table 946 Error Code Sub-objects Within a Business Group

The subobjects related to an EP (endpoint) can be those given in Table 947.

Code	Name	Description
0		Common EP errors
3	SIP	Dialing Plan errors

Table 947 Error Code Sub-objects Related to an EP

For Feature object, one sub-object is defined for each feature. Not all the sub-objects are defined now; more sub-objects will be defined as they are required.

Examples:

31111004 = subRC_APP_FEAT_ECF_START_TIME_INVALIDCategory: 3 = Application
Object: 11 = Feature
Sub-object: ECF
Attribute: Start time
Error: Invalid (last 3 digits can be anything)

30303009 = subRC_APP_BG_DEPT_BG_NAME_MISSING
Category: 3 = Application
Object: 05 = Business Group
Sub-object: 03 = Department
Attribute: BG name
Error: missing (last 3 digits can be anything)

10100001 = subRC_DB_SUB_DN_LOOKUP_FAILED
Category: 1 = Database
Object: 01 = Subscriber
Sub-object: N/A
Attribute: DN
Error: lookup failed (last 3 digits can be anything)

00000000 = SUCCESS

The old errors subRC_INTERNAL_ERROR and subRC_INTERNAL_RECOVERY_ERROR are now replaced by specific errors. The subRC_INTERNAL_RECOVERY_ERROR is replaced by subRC_DB_COMM_INTERNAL_RECOVERY_ERROR. All other errors beginning with 1 (database error), 2 (system error), and 9 (internal error) replace the old error subRC_INTERNAL_ERROR.

25.2 ResultMsgNumber

This field is not used.

25.3 Result Text Strings.

The Result Text fields contain additional error information. It is always returned.

ResultText2 through ResultText5 are defined in the table with the return codes. Only the fields applicable are returned.

ResultText1 contains the specific error. If no error (ReturnCode=0), then “Success” is returned.

ResultText2 contains the label of the field that was in error.

ResultText3 contains the value that was in error or the label of the field that was in conflict.

ResultText4 contains additional label that may be in error.

ResultText5 contains any internal error information

25.4 ResultTOY

The date and time that the functions are processed in the following format:

```
CYY-MM-DDThh: mm:ss
```

Example: 2002-09-12T12:35:30.

25.5 Result Codes/Result Text Settings

The list of possible return codes along with text describing the meaning of each code is provided in a deliverable file: /unisphere/srx3000/UNSPsubp/ReturnCodes.txt.

Note that an HTTP or SOAP error code will be issued if the XML request is syntactically invalid.

26 Using SOAP to Provision the OpenScape Voice Assistant

The following section outlines SOAP capabilities and functions that support the requirements of the OpenScape Voice Assistant V5 interface.

- Controlled Access to RTP Configuration Parameters
- Access to Call Statistics
- Backup and Restore via SOAP
- Provisioning of Time Destination Tables
- Provisioning of SIP Signaling Manager
- Provisioning of Billing (CDR)
- Tones and Announcements Management

For more information on the OpenScape Voice Assistant, refer to the *OpenScape Voice Assistant, Framework, Service Documentation; OpenScape Voice Assistant V2.0 Administrator Documentation; OpenScape Voice Assistant Feature Configuration Administration Guide;* and, the *OpenScape Voice Assistant API Description.*

26.1 RTP Configuration Management

The SOAP server is enhanced to have read/write access only to customer specific RTP parameters defined in the system. The RTP parameters that will be accessible by SOAP clients (access control list) are easy to configure and maintain. Soap Server's Configuration file (RtpParameters.conf) will assure that only particular RTP parameters are exported to the SOAP clients. If an RTP parameter is not included in this list then access is denied, even if this RTP defined as 'customer type' and is accessible via CLI or SNMP. This approach allows different configuration files to be provided for different solutions, i.e., OpenScape Voice. For a list of variables accessible via SOAP, refer to Table 949.

The structure of RtpParameters.conf file has been redefined in xml format, making parsing by Soap Server much more easier and efficient.

```
<RtpParameterList>

<RtpParameter>
    <Name>Srx/Service/CCS/CcsRecallTimerT4</Name>
    <Type>integer</Type>
    <Description>Timer T4 (Recall Supervision originating timer) is started
on sending a CCS recall indication to the CCBS/CCNR subscriber and is
stopped on receipt of an answer to this indication. Running on the Originating
```

```
side.</Description>
        <Default>30</Default>
        <Range>[15-60]</Range>
        <Length></Length>
        <Unit>seconds</Unit>
        <Restart>no</Restart>
        <Access>read/write</Access>
        <CustomGUI></CustomGUI>
</RtpParameter>

<RtpParameter>
        <Name>Srx/Sip/InternationalReroutingPAC</Name>
        <Type>string</Type>
        <Description>Number combination used for rerouting an international
call</Description>
        <Default>""</Default>
        <Range>[0-9],('',*,#)</Range>
        <Length>0-30</Length>
        <Unit>String</Unit>
        <Restart>no</Restart>
        <Access>read/write</Access>
        <CustomGUI></CustomGUI>
 </RtpParameter>

<RtpParameter>
        <Name>Srx/Sip/AuthEnabled</Name>
        <Type>boolean</Type>
        <Description>Flag to enable or disable Digest Authentication  </
Description>
        <Default>RtpFalse</Default>
        <Range>(true,false)</Range>
        <Length></Length>
        <Unit>N/A</Unit>
        <Restart>no</Restart>
        <Access>read/write</Access>
        <CustomGUI></CustomGUI>
 </RtpParameter>

…

</RtpParameterList>
```

The XML formatted RTP Parameters shall have to follow certain guidelines with regards to their syntax in order for SOAP Server to parse and validate key element tags.

For **integer** parameters:

- If it is a range of values it should be included in brackets [ ] e.g. [0-180]

- If it accepts specific values then those should be defined in parenthesis () separated by commas e.g. (1,2,3,6)

For **string** parameters:

- If it accepts a range of integer values it should be included in brackets [ ] e.g. [0-9]
- If it accepts a range of string values it should be included in curly brackets {} e.g. { A-E }
- If it accepts specific values then those should be defined in parenthesis () separated by commas e.g (*,#)
- Ranges and specific values can be combined, e.g. [0-9],(*,#) defines a field that accepts the numbers from 0 to 9 and the characters * and #.

For **boolean** parameters:

- The only expected value is (true,false).

It is imperative to note that if the syntax for the ValidRange and Length fields do not comply with the above rules, then the corresponding RTP Parameters shall not be subject to any validations.

With the new format in place, Soap Server Development are not responsible for the maintenance of the file, since the whole process has been automated. Production has delivered a script that parses the .parm files and generate the xml formatted RtpParameters.conf file. It is now the responsibility of the parm file owners to have an accurate and well defined description and header information.

Both of the GET and SET functions take a list of RTP parameters, which can be retrieved or assigned, and flags an error when the value to be "set" is outside the range of valid values.

SOAP Export supports the exporting command for the RTP parameters that are accessible via SOAP. The executable is invoked from within the RTP CLI (startCli) expert mode by using the command soapExport, followed by a string of parameters enclosed within double quotes. The syntax to perform this should be as follows:

From the expert mode prompt:

```
CLI>soapExport "-f=outputfilename  -RTP"
```

It is also possible to execute soapExport from the command line, as was supported in the past, but now this method is reserved for internal development and test purposes only, and therefore the operator ID and password parameters are required in addition to the parameters listed above. The command line syntax should be as follows:

```
./soapExport -o=Operator -pass=Password  -f=outputfilename
```

`-RTP`

where outputfilename is the location to which the exported RTP parameters is to be written.

The new massprov commands are designed to work in "MP2" mode and it should be identified by prefixing a ":MP2" at the end of the "FILE VERSION command" in the beginning of the command file. For example:

```
FILE VERSION:11.00.01:MP2
```

26.1.1 Set an RTP Parameter

SOAP Mass Provisioning is impacted to support the RTPSet command for the RTP parameters that are accessible via SOAP. To Set/Update an RTP parameter on the system, use the **#RTPSET** command to retreive a list of the RTP parameters that are available via SOAP. Next, the input must contain the name of the RTP parameter (e.g. hiQ/CSTA/ListenerPortNumber) and the value. For example:

RTPSET

- ,RTPNAME="Name of the RTP parameter"
- ,RTPVALUE="Value of the RTP parameter"

SoapServer will not be responsible for the validation of the passed parameter values. The only validation that is done is whether the RTP Parameter name that has been passed in the request exists in the configuration file.

26.1.2 Display the List of Accessible RTP Parameters

To retrieve an existing RTP parameter on the system, a list of parameters must be input containing the name of the RTP parameter (e.g. hiQ/CSTA/ ListenerPortNumber).

SoapServer will validate that the requested parameter exists in the configuration file and will return to the user the parameter name and the value currently assigned to this parameter.

The existing subGetRtpParm common function will be used in order to obtain a coPaging, Sorting (Sorting Direction: Ascending, Descending) and Filtering based on RTP Parameter Name with/without wildcard support shall also be supported. It is important to note that the only wildcard character to be supported is "*", and it should only be at the end of RTP Parameter Name to be retrieved. For example: If the administrator wishes to retrieve all the SIP RTP Parameters, the filter on the Assistant screen could be something like: Srx/Sip/*

With regards to RTP Parameters having dedicated GUI, the GetConfigParmDataResult interface shall only return those, only if the Name field on the GetConfigParmDataRequest includes the full name of the Parameter. The latter implies that the administrator is aware of the specific RTP parameter he/she wants to retrieve.

Configuration parameter from the RTP configuration table.

Table 948 outlines the available interfaces to support accessing the RTP database.

Field Name	Required	Description	Type	Max Length (Range)	Default Value
OperatorId	Yes	I/F operator	Char array	40 (1…40)	N/A
ConfigDataNameList	No	I/F Parameter Name	Char array	N/A	N/A
GetDedicated GUIParameters	No	Flag indicating whether Dedicated GUI Parameters should be returned	Boolean	True/False	False
ParmMismatchFilter	No	Option for retrieving Parameters whose value is the same or different from the default/ suggested one	Enumerator	Range of Values: Fetchall, Fetchmismatched FetchNonMismatched	FetchAll
Sorting	No	The sort order of the retrieved result set	Enumerator	Ascending Descending	Ascending
Page Number	No	number of the retrieved result set	Integer	-	-
Page Size	No	Page size of the retrieved result set	Integer	-	-

Table 948 "GetConfigParmDataRequest Input Field definitions

Table 949 lists the variables accessible via SOAP.

hiQ/CSTA/CSTAManger1VirtualIp	Srx/Service/IN/MaxSerialTriggerCount
hiQ/CSTA/CSTAManger2VirtualIp	Srx/Service/IN/N11TriggerOfficeOption
hiQ/CSTA/IntervalTimeForReplication	Srx/Service/IN/ONoAnswerTimer
hiQ/CSTA/ListenerPortNumber	Srx/Service/IN/SDSTriggerOfficeOption
hiQ/CSTA/MaxCSTAConnections	Srx/Service/IN/SendToResourceTimer
hiQ/CSTA/MaxMonitoredEndpoints	Srx/Service/IN/T1Timer
hiQ/TraffMeas/Settings15MinRetPeriod	Srx/Service/IN/T1Timer
hiQ/TraffMeas/Settings30MinRetPeriod	Srx/Service/IN/TNoAnswerTimer
hiQ/TraffMeas/Settings5MinRetPeriod	Srx/Service/INAP/DP4CauseValues
hiQ/TraffMeas/SettingsDailyRetPeriod	Srx/Service/INAP/DP5CauseValues
hiQ/TraffMeas/SettingsHourlyRetPeriod	Srx/Service/INAP/ObjectIdentfierSCFSSF
Srs/Auth/AuthenticationStrategy	Srx/Service/INAP/ObjectIdentfierSSFSCF
Srs/Auth/DatabaseProcess	Srx/Service/INAP/SII2AllowCdlNNoPresentationIndicator
Srx/IpPacketPriority/DSCPTOS/SIP/Type	Srx/Service/INAP/SII2BothwayThroughConnectionIndicator
Srx/IpPacketPriority/DSCPTOS/SIP/Value	Srx/Service/INAP/SII2BwdCallCompletionTreatmentIndicator
Srx/Main/ACMChargeIndication	Srx/Service/INAP/SII2BwdConferenceTreatmentIndicator
Srx/Main/ACRCfmTreat	Srx/Service/INAP/SII2ConnectedNumberTreatmentIndicator
Srx/Main/ApplicationType	Srx/Service/INAP/SII2FwdCallDiversionTreatmentIndicator
Srx/Main/BillingMethod	Srx/Service/INAP/SII2FwdCallingPartyRestrictionIndicator
Srx/Main/CallingPartyDisplayPrependPrefix	Srx/Service/INAP/SII2FwdCallOfferingTreatmentIndicator
Srx/Main/CallingPartyPrefixDigitString	Srx/Service/INAP/SII2FwdConferenceTreatmentIndicator
Srx/Main/CCCDF/IpAddress	Srx/Service/INAP/SII2SupressCallDiversionNotification
Srx/Main/CCCDF/PortNumber	Srx/Service/INAP/SII2SupressCallTransferNotification
Srx/Main/CFMaxDiversions	Srx/Service/INAP/SII2SuspendTimer
Srx/Main/ClgPtyDisplayPrependPNAC	Srx/Service/INAP/SII2UserDialogueDurationIndicator
Srx/Main/CorrelationInfoUsage	Srx/Service/INAP/T_Long
Srx/Main/Country	Srx/Service/INAP/T_Medium
Srx/Main/CpuRingListMaxEntries	Srx/Service/INAP/T_Short
Srx/Main/CustomerType	Srx/Service/INAP/T_SrfTimeout
Srx/Main/cwtOperMethod	Srx/Service/INAP/T_SsfErr
Srx/Main/hiQVariant	Srx/Service/INAP/T_SsfOpSeq
Srx/Main/OutGoingCallingPartyNumberType	Srx/Service/INAP/T_SsfWfInstructions
Srx/Main/OwnCountryCode	Srx/Service/INAP/T_SsfWfUserInteraction
Srx/Main/PrefixDigitStringToCallingParty	Srx/Service/INAP/TCAPManagerName
Srx/Main/RACFCfmTreat	Srx/Service/INAP/TDP3TriggerOfficeOption
Srx/Main/SrxCallingNameDeliveryMethod	Srx/Service/INAP/TDP4TriggerOfficeOption

Table 949 RTP Parameter Path / Name (Sheet 1 of 4)

Srx/Main/SrxCallWaitingTimer	Srx/Sip/AlertTimer
Srx/Main/SrxDefaultLang	Srx/Sip/AuthClientQopPreference
Srx/Main/SrxLocalCallingNameDelivery	Srx/Sip/AuthEnabled
Srx/Main/SrxSupportedLangList	Srx/Sip/AuthGenerateNextNonce
Srx/Main/SrxSwitchLangMode	Srx/Sip/AuthMaxAttempts
Srx/Main/SS7Variant	Srx/Sip/AuthNonceLifetime
Srx/Main/SvcAssignedPerOfficeACR	Srx/Sip/default_number_of_sessions
Srx/Main/SvcAssignedPerOfficeCCW	Srx/Sip/HungupTimer
Srx/Main/SvcAssignedPerOfficeCIDS	Srx/Sip/InternationalReroutingPAC
Srx/Main/SvcAssignedPerOfficeCNAB	Srx/Sip/IntlE164Numbers
Srx/Main/SvcAssignedPerOfficeCNDB	Srx/Sip/IpPort
Srx/Main/SvcAssignedPerOfficeLINR	Srx/Sip/IpPortTLS
Srx/Main/SvcAssignedPerOfficeLONR	Srx/Sip/LocalReroutingPAC
Srx/Main/SvcAssignedPerOfficeMCT	Srx/Sip/LongCall_disconnect_enabled
Srx/Main/SvcAssignedPerOfficeUSCOLR	Srx/Sip/LongCallTimer
Srx/Main/SvcAssignedPerOfficeUSTWC	Srx/Sip/max_number_of_sessions
Srx/Main/SvcInternationalPrefix	Srx/Sip/Min_Expires_Dialog
Srx/Main/SvcNationalPrefix	Srx/Sip/Min_Expires_Keyset
Srx/Main/twcOperMethod	Srx/Sip/Min_Expires_Presence
Srx/Service/ACAR/ACFeatureAssignedPerOffice	Srx/Sip/Min_Expires_Publish
Srx/Service/ACAR/ AnnouncementResponseTimeoutCounter	Srx/Sip/Min_Expires_Toggle
Srx/Service/ACAR/AnnouncementResponseTimer	Srx/Sip/Min_Session_Timer_Value
Srx/Service/ACAR/AnnouncementResponseTimer	Srx/Sip/NationalReroutingPAC
Srx/Service/ACAR/ARActivationProcedure	Srx/Sip/ReleaseAckTimer
Srx/Service/ACAR/ARActivationProcedure	Srx/Sip/ReroutingEndpointEnabled
Srx/Service/ACAR/ARFeatureAssignedPerOffice	Srx/Sip/ReroutingSubscriberEnabled
Srx/Service/ACAR/ARFeatureAssignedPerOffice	Srx/Sip/ReroutingTimerInvite
Srx/Service/ACAR/DenyARActivationsToAnonymousDN	Srx/Sip/ReroutingTimerTrying
Srx/Service/ACAR/DenyARtoNonUniqueDN	Srx/Sip/SDPProtocol
Srx/Service/ACAR/DNVoiceBackOption	Srx/Sip/Session_Timer
Srx/Service/ACAR/InitialQueryResponseTimer	Srx/Sip/SipUpdateSupported
Srx/Service/ACAR/InterBusyIdleQueryDurationTimer	Srx/Sip/Timer_T1
Srx/Service/ACAR/InvalidDialedDigitCounter	Srx/Sip/Timer_T2
Srx/Service/ACAR/MaxNum6SecondSpecialRingingCycles	Srx/Sip/Timer_T4
Srx/Service/ACAR/ MaxNumConcurrentACActivationsAllowedPerCustomer	Srx/Sip/UceGenericTimer

Table 949 RTP Parameter Path / Name (Sheet 2 of 4)

Srx/Service/ACAR/ MaxNumConcurrentARActivationsAllowedPerCustomer	Srx/Sip/UdpPort
Srx/Service/ACAR/ MaxNumQueueEntriesAllowedPerCalledParty	Srx/Sip/VMServerIPAddress
Srx/Service/ACAR/ MaxNumUnansweredSpecialRingingApplications	Srx/Sip/VMSubscriptionTime
Srx/Service/ACAR/MonitoringTimer	Srx/Subp/Address
Srx/Service/ACAR/OriginatingOverallMonitoringTimer	Srx/Subp/Address
Srx/Service/ACAR/OriginatingScanRateTimer	Srx/Subp/ClearErrorDays
Srx/Service/ACAR/OutstandingNotificationTimer	Srx/Subp/ClearJobDays
Srx/Service/ACAR/PeriodicScanRate	Srx/Subp/DebugGsoap
Srx/Service/ACAR/PerSPCSBlkARMsgWICNInfo	Srx/Subp/JobCheckSleep
Srx/Service/ACAR/RequestTerminatingScanningAllowed	Srx/Subp/JobClearTime
Srx/Service/ACAR/ResumeScanningThresholdTimer	Srx/Subp/JobLoopSleepTime
Srx/Service/ACAR/ResumeScanningTimer	Srx/Subp/NumberOfInstances
Srx/Service/ACAR/SubsequentQueryResponseTimer	Srx/Subp/NumberOfInstances
Srx/Service/ACAR/TerminatingScanningMonitoringTimer	Srx/Subp/NumObjectsToPauseAfter
Srx/Service/CCS/CCBSFeatureAssignedPerOffice	Srx/Subp/OverLoadCheckPeriod
Srx/Service/CCS/CcbsRequestOperationT2	Srx/Subp/OverLoadLevel1Sleep
Srx/Service/CCS/CcbsScanRateT10	Srx/Subp/OverLoadLevel2Sleep
Srx/Service/CCS/CcbsServiceDurationT3	Srx/Subp/OverLoadLevel3Sleep
Srx/Service/CCS/CcbsServiceSupervisionT7	Srx/Subp/Port
Srx/Service/CCS/CCNRFeatureAssignedPerOffice	Srx/Subp/Port
Srx/Service/CCS/CcnrRequestOperationT2	Srx/Subp/Priority
Srx/Service/CCS/CcnrScanRateT10	Srx/Subp/SubSchedPriority
Srx/Service/CCS/CcnrServiceDurationT3	Srx/Subp/TombstoneKeepRecordsInterval
Srx/Service/CCS/CcnrServiceSupervisionT7	Srx/Usc/AutoDelete
Srx/Service/CCS/CcsIdleGuardT8	Srx/Usc/EncryptedBackupPassword
Srx/Service/CCS/CcsIdleGuardT8	Srx/Usc/EncryptedPassword
Srx/Service/CCS/CcsRecallT9	Srx/Usc/FtpPort
Srx/Service/CCS/CcsRecallTimerT4	Srx/Usc/Host
Srx/Service/CCS/ MaxNumInCCBSActivationsAllowedPerCustomer	Srx/Usc/HostBackup
Srx/Service/CCS/ MaxNumInCCNRActivationsAllowedPerCustomer	Srx/Usc/Method
Srx/Service/CCS/ MaxNumOutCCBSActivationsAllowedPerCustomer	Srx/Usc/Password
Srx/Service/CCS/ MaxNumOutCCNRActivationsAllowedPerCustomer	Srx/Usc/PasswordBackup
Srx/Service/CCS/RetentionTimer	Srx/Usc/Refresh

Table 949 RTP Parameter Path / Name (Sheet 3 of 4)

Srx/Service/CFBlockingActive	Srx/Usc/Rename_Extension
Srx/Service/CFBlockingCodes	Srx/Usc/Retention_Period
Srx/Service/IN/FinalTreatmentName	Srx/Usc/Second_Storage
Srx/Service/IN/FinalTreatmentName	Srx/Usc/User
Srx/Service/IN/FinalTreatmentType	Srx/Usc/UserBackup
Srx/Service/IN/FinalTreatmentType	Srx/Main/node1CIPLogical
Srx/Service/IN/INServiceSSN	Srx/Main/node2CIP0
Srx/Main/ClusterName	Srx/Main/node2CIP1
Srx/Main/node1Name	Srx/Main/node2Logical
Srx/Main/node1PhyIpAddr	Srx/Main/node1CIPNetmask
Srx/Main/node2Name	Srx/Main/node1CIPNetwork
Srx/Main/node2PhyIpAddr	Srx/Main/node2CIPNetmask
Srx/Main/ClusterType	Srx/Main/node2CIPNetwork
Srx/Main/node1CIP0	Srx/Main/CIPGateway
Srx/Main/node1CIP1	Srx/Main/ciControlTos
Srx/Main/CIPBroadcast	Srx/Lic/DynamicWarningThres
Srx/Main/CIP2Gateway	Srx/Irm/AudioPacketInterval_AAC_LC
Srx/Main/ciDataTos	Srx/Irm/AudioPacketInterval_G7221_24
Srx/Main/UseCTSecurityInterceptDest	Srx/Irm/AudioPacketInterval_G7221_32
Srx/Irm/AudioBitRate_AAC_LC	Srx/Irm/AudioPacketInterval_G7221_48
Srx/Irm/AudioBitRate_G7221_24	
Srx/Irm/AudioBitRate_G7221_32	
Srx/Irm/AudioBitRate_G7221_48	

Table 949 RTP Parameter Path / Name (Sheet 4 of 4)

26.2 Call Statistics

This interface allows a SOAP client to have read-only access to all statistics counters defined in the RTP database.

Attention: The interface is generic; new statistics counters that will be introduced in the future will not have further SOAP impacts.

To attain access to the call statistics counters in the RTP database, the user provides:

- **GetCounterInfo**: enter the name of the counter

- **GetCounterList**: This function returns an array, containing all counter names from the database, and facilitates paging through the result set by SOAP clients that display the data graphically to the user. Therefore the following parameters are also included:
  - Page Size
  - Page Number

The response includes the following data, for use by the SOAP client in paging:

- Number of rows in the result set (total number of Endpoint entries)
- Number of pages in the result set (total number of pages of data, given the number of rows and the Page Size)
- Page Size (same as Page Size in request)
- Page Number (same as Page Number in request)

The paging parameters are optional, and if not passed, the behavior will default to returning all existing Endpoints with the maximum of 1000 entries. For each entry, the Statistics Counter Name is returned. For more information on Paging, refer to Section 5.5, “Paging”, on page 72.

SoapServer will use the existing C Interface provided by RTP to support the above functionality as described in the sections below.

26.2.1 Display the Counter Value

To get the value of a counter from the database, the user inputs the following commands:

```
RtpReturn_t
RtpStaGetCounterValue
```

The output shows:

```
const RtpChar *counterName,RtpInt32 *value, const
RtpBool_t flush
```

Parameter definitions:

- **counterName**: Counter ID unique throughout the cluster. May only contain letters, digits or underscores.
- **value**: Pointer to a variable of type RtpInt32 to be used to return the counter value.
- **flush**: If set to a value other than null, this parameter causes local increments/decrements to be flushed before returning the counter value.

26.2.2 Display the Counter Status

To get the status of a counter from the database, the user inputs the following commands:

```
RtpReturn_t
RtpStaGetCounterStatus
```

The output shows:

```
const RtpChar *counterId,RtpStaCounterStatus_p stat
```

Parameter definitions:

- **countered**: Counter ID unique throughout the cluster.
- **Stat**: Pointer to a structure of type RtpStaCounterStatus_t:
  - typedef struct {
  - RtpBool_t stat_status;
  - RtpInt32 stat_flushint;
  - RtpStaCounterStatus_t, *RtpStaCounterStatus_p;

If successful, the element stat_status will contain the value RtpTrue if the counter is activated, else RtpFalse. The element stat_flushint will contain the value of the flush interval specified when the counter was activated (cf. RtpStaActivateCounter).

26.2.3 Display the Counter Names

To retrieve the list of names of all counters from the database, the user inputs the following commands:

```
RtpReturn_t
RtpStaGetCounters
```

The output shows:

```
RtpStaCounterList_t **counterList
```

Parameter definitions:

- **counterList**: Address of a pointer used to store the location of the counter names array.

Table 950 outlines the statistics counters defined in the RTP database.

Return Value	Function	Description
struct unsp__GetCounterValueResult	int unsp__GetRTPCounterInfo (struct unsp__GetCounterRequest GetCounterValueRequest, struct unsp__GetCounterValueResult *GetCounterValueResult)	This function returns the value of a counter from the database.
struct unsp__GetCounterStatusResult	int unsp__GetRTPCounterStatus (struct unsp__GetCounterRequest GetCounterStatusRequest, struct unsp__GetCounterStatusResult *GetCounterStatusResult)	This function returns the status of a counter from the database.
struct unsp__GetCounterListResult	int unsp__GetRTPCounterList (struct unsp__GetCounterListRequest GetCounterListRequest, struct unsp__GetCounterListResult *GetCounterListResult)	This function returns an array containing all counter names from the database.

Table 950 Call Statistic Counters

26.3 Database Backup and Restore

SoapServer is enhanced to provide interfaces to support the following database operations:

- Start a backup process in one or both nodes;
- Cancel a backup;
- Query for the status of the backup process;
- Start a restore process;
- Cancel a restore;
- Query for the status of the restore process.

The existing functionality provided by the Maintenance Manager is used in order to initiate, cancel or query for the status of maintenance tasks such as the backup and restore. The Maintenance manager provides the following APIs to support this:

- RequestHandler.createJob
- RequestHandler.cancelJob
- RequestHandler.getJobs

26.3.1 Create Database Job Request

To create a job in order to start the Backup or Restore process, the user is prompted to enter the following information:

- **Job Type**: which indicates the required action. Possible values will be backup and restore
- **Arguments** [up to 256 char]: which indicated the type of the backup or restore and can be any of the following values {all, oracle_rtp, oracle_srx, oracle_all, omni, solid}.

The response will include the JobID that has been assigned to the request and can be used later in order to cancel or query this specific Job.

26.3.2 Cancel Database Job Request

The Backup and Restore processes may be cancelled while they are running. The user may cancel a job request by entering the required JobId. Not all jobs can be cancelled. To determine if a job can be aborted, the “cancellable” attribute will appear in the Job Definition's Type field.

26.3.3 Get Database Job List

The user may retrieve a list of all created jobs, completed and running.This new interface facilitates paging through the result set by SOAP clients that display the data graphically to the user. For more information on Paging, refer to Section 5.5, “Paging”, on page 72. Additionally, the paging request will also accept, as optional input, the JobID in order to filter the expected results on the specific job.

26.3.4 Values for Job Request

For each entry, the values in Table 951 are returned.

Parameter	Description
Job ID	A cluster-unique index number assigned when a job is created.
Job Name	A unique alphanumeric string identifying the job.
Executable Path	Indicates the path to the job’s executable, relative to the job definition file.
Arguments (args)	Indicates the format of the argument required for a particular job. The maximum length is 256 characters.

Table 951 Job Request Values (Sheet 1 of 2)

Parameter	Description
State	Indicates whether the job is: • 1: running - running • 2: overdue - running, but past max time • 3: cancelling - running, attempting to cancel • 4: cancelled - terminated, cancelled • 5: done - terminated, success • 6: error - terminated, error
Type	A Bitmask that describes the job attributes. The type may contain one or more type attributes such as: • BASIC - job contains no attributes • CANCELABLE - job may be cancelled • SCHEDUALABLE - job may be scheduled • BLOCK_ON_DONE - job is blocked when completed • BLOCK_ON_ERR - job is blocked when an error occurs • TRAP_ON_ERR - trap is sent when job ends in error • WRITES_LOG - log file is written upon completion of job Example: WRITES_LOG | CANCELABLE |BLOCK_ON_ERR | TRAP_ON_ERR
Average Run Time	Indicates the estimated average run time (in seconds) required for a job to complete.
Maximum Run Time	Indicates the estimated maximum run time (in seconds) required for a job to complete.
Report Name	Indicates the path and name of the report generated for this job. The information contained in a Report is brief.
Log Name	Indicates the path and name of the log generated for this job. The information contained in a Log is detailed. **NOTE**: This is a future feature.
Start Time	Indicates the start time of the job.
Stop Time	Indicates the stop time of the job.

Table 951 Job Request Values (Sheet 2 of 2)

26.4 Managing Time Destinations Tables

SoapServer now supports the managing of time destination, day, period, and weekly schedules on the system.

26.4.1 Time Destination

To create or remove a time destination, the user is prompted to provide:

OperatorID: The name of the user operator used for logging purposes.

Time Destination Name:The name of the Time Destination, a string of maximum 15 characters.

To display a Time Destination, the user can optionally input the “Time Destination Name” in which case only that destination is retrieved otherwise the command retrieves all time destinations from the database, and facilitates paging through the result set by SOAP clients that display the data graphically. For more information on Paging, refer to Section 5.5, “Paging”, on page 72.

Table 952 outlines the interfaces to support the provisioning of the Time Destinations functionality.

Return Value	Function
struct unsp_CreateTimeDestinationResult	int unsp__createTimeDestination (unsp__CreateTimeDestination CreateTimeDestinationRequest, struct unsp__CreateTimeDestinationResult *CreateTimeDestinationResult);
struct unsp__RemoveTimeDestinationResult	int unsp__removeTimeDestination (unsp__RemoveTimeDestination RemoveDestinationRequest, struct unsp__RemoveTimeDestinationResult * RemoveTimeDestinationResult);
struct unsp__TimeDestinationResult	int unsp__queryTimeDestination (unsp__QueryTimeDestinationRequest QueryTimeDestinationRequest, struct unsp__TimeDestinationResult * QueryTimeDestinationResult);

Table 952 Time Destination Interfaces

26.4.2 Day Schedule

The Day Schedule is used to provision a specific day schedule within the time destination. The schedule that is assigned within the time destination should cover one whole week.

SoapServer provides interfaces for create, assign, unassign, delete, and display operations

- To create a Day Schedule, the user is prompted to provide:
  - **Time Destination Name**: Identifying the name of an existing Time Destination on the system
  - **Day Schedule Name**: Identifying the name of the Day Schedule object. This object uniquely identifies a day schedule within the time destination.
- To assign a Day Schedule to a Time Destination, the user is prompted to provide:
  - **Time Destination Name**: Identifying the name of an existing Time Destination on the system

  - **A list of Day Schedules**, each one specified for a day of the week, may be specified by the Day Schedule Name.
  - Any day schedule that is assigned must be fully allocated, meaning that it must contain period schedules that cover exactly 24 hours (1440 minutes).
- To unassign a Day Schedule from a Time Destination on the specified days, the user is prompted to provide:
  - **Time Destination Name**: Identifying the name of an existing Time Destination on the system
  - **A Boolean value for every day of week**. If it is marked true, the day schedule on that day is unassigned
- To modify Day Schedules of Time Destination the user is prompted to provide:
  - **Time Destination Name**: Identifying the name of an existing Time Destination on the system
  - **A list of Day Schedules** may be specified with Day Schedule Name values, and the days of the week.
- To remove Day Schedules from Time Destination, the user is prompted to provide:
  - **Day Schedule Name**: Identifying the name of the Day Schedule object to be removed.
  - **Time Destination Name**: Identifying the name of an existing Time Destination on the system
  - The command removes the day schedule and all of its period schedules, provided it is not assigned to a time destination.

Table 953 outlines the interfaces to support the provisioning of the Day Schedule functionality.

Return Value	Function
struct unsp__CreateDayScheduleResult	int unsp__createDaySchedule (unsp__CreateDayScheduleRequest CreateDayScheduleRequest, struct unsp__CreateDayScheduleResult * CreateDayScheduleResult)
struct unsp__ModifyDayScheduleResult	int unsp__modifyDaySchedule (unsp__ModifyDayScheduleRequest ModifyDayScheduleRequest, struct unsp__ModifyDayScheduleResult * ModifyDayScheduleResult)

Table 953 Day Schedule Interfaces (Sheet 1 of 2)

Return Value	Function
struct unsp__RemoveDayScheduleResult	int unsp__removeDaySchedule (unsp__RemoveDayScheduleRequest RemoveDayScheduleRequest, struct unsp__RemoveDayScheduleResult * RemoveDayScheduleResult)
struct unsp__DayScheduleResult	int unsp__queryDaySchedule (unsp__QueryDayScheduleRequest QueryDayScheduleRequest, struct unsp__DayScheduleResult * QueryDayScheduleResult)

Table 953 Day Schedule Interfaces (Sheet 2 of 2)

26.4.3 Period Schedule

The Period Schedule table is used to provision the time period within a day schedule. The time period is in the hours:minutes (HH:MM) format and must cover a 24 hour period.

- To create/modify a Period Schedule, the user is prompted to provide:
  - **Day Schedule Name** to which it belongs and is identified by dayName
  - **Time Destination Name** identified by timeDestName
  - **starting time** of the period is designated by a character string startTime (such as 06:00)
  - the **stop time** of the period is designated by a character string stopTime (such as 11:59)
  - **Generic destination** with which this period is associated is identified by destName
- To remove a Period Schedule, the user is prompted to provide:
  - **Day Schedule Name** to which it belongs and is identified by dayName
  - **Time Destination Name** identified by timeDestName
  - **startTime** of the period
  - The remove command deletes the period schedule provided its parent day schedule is not *explicitly* assigned to a time destination with the Assign Day Schedule command.
- To display a Period Schedule, the user is prompted to provide the following optional parameters:
  - Day Schedule Name to which it belongs
  - Time Destination Name

- startTime of the period
- Page Size
- Page Number

In all cases, if the time destination is given and/or the day schedule is given, they act as a simple filter on the set of period schedules that may be retrieved. If a specific period schedule object is identified by the combination of the day schedule and the starting time, the day schedule is not used as a filter. For each entry, the Day Schedule Name, Time Destination Name , stopTime, stopTime and Destination Name will be returned.

For more information on Paging (data responses), refer to Section 5.5, “Paging”, on page 72.

- To display a Weekly Schedule, the user is prompted to provide the following parameters:
  - Time Destination Name
  - A Day for which a schedule may be assigned (optional)

Weekly schedules include the:

- Time Destination;
- day for which the period is assigned
- day schedule that is assigned to (Time Destination, day).

Attention: Day schedules may appear more than once, particularly if the day schedule is assigned to more than one day for a destination.

Table 954 outlines the interfaces to support the provisioning of the Period and Weekly Schedule functionality.

Return Value	Function
struct unsp__CreatePeriodScheduleResult	int unsp__createPeriodSchedule (unsp__CreatePeriodScheduleRequest CreatePeriodScheduleRequest, struct unsp__CreatePeriodScheduleResult * CreatePeriodScheduleResult)
struct unsp__ModifyPeriodScheduleResult	int unsp__modifyPeriodSchedule (unsp__ModifyPeriodScheduleRequest ModifyPeriodScheduleRequest, struct unsp__ModifyPeriodScheduleResult * ModifyPeriodScheduleResult)

Table 954 Period and Weekly Schedule Interfaces (Sheet 1 of 2)

Return Value	Function
struct unsp__RemovePeriodScheduleResult	int unsp__removePeriodSchedule (unsp__RemovePeriodScheduleRequest RemovePeriodScheduleRequest, struct unsp__RemovePeriodScheduleResult * RemovePeriodScheduleResult)
struct unsp__PeriodScheduleResult	int unsp__queryPeriodSchedule (unsp__QueryPeriodScheduleRequest QueryPeriodScheduleRequest, struct unsp__PeriodScheduleResult * QueryPeriodScheduleResult)
struct unsp__WeeklyScheduleResult	int unsp__queryWeeklySchedule (unsp__QueryWeeklyScheduleRequest QueryWeeklyScheduleRequest, struct unsp__WeeklyScheduleResult * QueryWeeklyScheduleResult)

Table 954 Period and Weekly Schedule Interfaces (Sheet 2 of 2)

26.5 SIP Signaling Manager

SoapServer is enhanced to support the provisioning of the SIP_Trusted_Entities table that extends the existing System_SIP_Profile table. New interfaces allow the assigning and unassigning of Port Ranges to/from SIP Entities and also to display a list of the port ranges which are assigned to a SIP entity.

26.5.1 Assign Port Range

To assign a port range, the user is prompted to provide:

- The IP address of the SIP entity, and
- A list of port ranges in which each entry will be defined of a Minimum Port and a Maximum Port.

If the minimum port value [0...65535] has a value equal to "0", then all the range of ports for this IP is defined as trusted.

If the minimum port value is > 0 and maximum port value > 0 then all the ports, from MIN_PORT to MAX_PORT, are defined as trusted entities.

If the maximum port value [0...65535] has a value equal to "0", this means that the minimum port value is defined as a single trusted port for this IP.

Since the user provides a list of Port Ranges there might be cases that a number of ports are not successfully assigned to SIP Entity. SoapServer will skip those Ports that cannot be assigned and, in terms of error checking, the response to the user will include the list of Port Ranges (MinPort/MaxPort) that were not successfully assigned to SIP Entity.

26.5.2 Unassign Port Range

To unassign a port range, the user is prompted to provide:

- The IP address of the SIP entity, and
- A list of port ranges that the user wants removed from the SIP Entity defined in a set of a Minimum Port and a Maximum Port.

26.5.3 Display Port Ranges

The user is prompted to provide the following optional parameters:

- IP Address,
- Page Size,
- Page Number.

If the user provides the IP Address of a SIP Entity, then the entire list of the port ranges, which are assigned to this entity, is displayed. Otherwise, the entire list of IP's is displayed with the corresponding Port Ranges. For each entry, the list of Port Ranges (MinPort and MaxPort) is returned.

For more information on Paging, refer to Section 5.5, "Paging", on page 72.

Table 955 outlines the interfaces to support the provisioning of the SIP Trusted Entity functionality.

Return Value	Function
struct unsp__AddSipTrustedEntityPortsResult	int unsp__AddSipTrustedEntityPorts (struct unsp__SipTrustedEntityPortsRequest AddSipTrustedEntityPortsRequest, struct unsp__AddSipTrustedEntityPortsResult *AddSipTrustedEntityPortsResult)
struct unsp__RemoveSipTrustedEntityPortsResult	int unsp__RemoveSipTrustedEntityPorts (struct unsp__SipTrustedEntityPortsRequest RemoveSipTrustedEntityPortsRequest, struct unsp__RemoveSipTrustedEntityPortsResult * RemoveSipTrustedEntityPortsResult)
struct unsp__GetSipTrustedEntityPortsListResult	int unsp__GetSipTrustedEntityPortsList (struct unsp__GetSipTrustedEntityPortsListRequest GetSipTrustedEntityPortsListRequest, struct unsp__GetSipTrustedEntityPortsListResult* GetSipTrustedEntityPortsListResult)

Table 955 SIP Trusted Entities Interfaces

26.6 Billing and CDR Generation

The generic RTP interface maintains a list of RTP parameters together with their range of valid values in a file called RtpParameters.conf. The 3 new RTP parameters and their valid ranges will be added to this .conf file.

These three new parameters permit the modification and display of CDR usage. This new CDR configuration management capability enables OpenScape Voice Accounting Management to perform basic call accounting (incoming, outgoing and internal calls) as well as ensure correct CDR generation and call duration for Enterprise call transfer scenarios for all possible customer configurations.

Srx/Usc/nodeID_area0	The areaID value that comprises the nodeID variable. Integer 0...999.
Srx/Usc/nodeID_area1	The areaID value that comprises the nodeID variable. Integer 0...255.
Srx/Usc/nodeID_area2	The areaID value that comprises the nodeID variable. Integer 0...31.

26.7 Tones and Announcements

SoapServer supports the provisioning of Tones and Announcements on the system.

26.7.1 Intercepts

An intercept is a sequence of treatments that can be repeated for a certain number of cycles. Ordered treatments for calls cannot be completed as dialed. These calls are redirected to a recorded announcement, a tone, an operator, or combinations thereof.

26.7.1.1 Create an Intercept

To create an intercept, the user is prompted to provide the following parameters:

- **Intercept Name**: The Intercept Name is an alphanumeric string [1..15] identifying an Intercept entry.
- **Default Intercept Name**: The Default Intercept Name is an alphanumeric string [1..15] identifying a Default Intercept entry. If a Default Intercept Name is assigned, then all of its corresponding assigned treatments also become the treatments used, unless they are modified to be different.

26.7.1.2 Modify an Intercept

To modify an intercept, the user is prompted to provide the following parameters:

- **Intercept Name**: The Intercept Name is an alphanumeric string [1..15] identifying an Intercept entry.
- **Cycle:** The Cycle represents the number of times that the Intercept Treatment sequence is applied. This Cycle object is the same for all treatments assigned to the same Intercept. A treatment reference must exist in order to modify this object
- **Interval:** The Interval represents silence between cycles. The Interval must be the same for all treatments assigned to the same intercept. A default of '0' interval is interpreted as '0000' milliseconds. A treatment reference must exist in order to modify this object
- **Default Intercept Name**: The Default Intercept Name is an alphanumeric string [1..15] identifying a Default Intercept entry. If a Default Intercept Name is assigned, then all of its corresponding assigned treatments also become the treatments used, unless they are modified to be different.

26.7.1.3 Remove an Intercept

To remove an intercept, the user is prompted to provide the following parameter:

- **Intercept Name**: The Intercept Name is an alphanumeric string [1..15] identifying an Intercept entry.

26.7.1.4 Display an Intercept

To display an intercept, the user is prompted to provide the following optional parameters:

- Intercept Name,
- Page Size,
- Page Number.

If the Intercept Name is given, it acts as a simple filter on the set of Intercepts that may be retrieved. For each entry, the Intercept Name, Cycle, Interval, Default Intercept Name, and Number of Intercepts is returned. For more information on Paging, refer to Section 5.5, “Paging”, on page 72.

Table 956 outlines the interfaces to support the provisioning of intercepts.

Return Value	Function
struct unsp__CreateInterceptResult	int unsp__createIntercept (unsp__CreateInterceptRequest CreateInterceptRequest, struct unsp__CreateInterceptResult * CreateInterceptResult)
struct unsp__RemoveInterceptResult	int unsp__removeIntercept (unsp__RemoveInterceptRequest RemoveInterceptRequest, struct unsp__RemoveInterceptResult * RemoveInterceptResult)
struct unsp__ModifyInterceptResult	int unsp__modifyIntercept (unsp__ModifyInterceptRequest ModifyInterceptRequest, struct unsp__ModifyInterceptResult * ModifyInterceptResult)
struct unsp__QueryInterceptDataResult	int unsp__queryIntercept (unsp__QueryInterceptRequest QueryInterceptRequest, struct unsp__QueryInterceptDataResult * QueryInterceptResult)

Table 956 Intercept Interfaces

26.7.2 Treatments

A treatment of an intercept can be defined as a tone, or an announcement. Each tone is identified by a system defined ID as well as a duration period.

26.7.2.1 Assign a Treatment

To assign a treatment to an existing intercept the user is prompted to provide the following parameters:

- **Treatment Order**: Describes the desired ordering of the new treatment to be created in relation to existent treatments. Only three (3) treatments are allowed per intercept. This is a mandatory object for creation. It can be any value of {First, Second, Third}.
- **Treatment type**: This mandatory field can be any value from {'Tone, Announcement,RelCall}.
  - If **Tone** is chosen, then the user should provide:
    - Tone Name field is mandatory. Tone Name is an alphanumeric string [0..15] identifying a Tone.
    - Duration, used in setting a length for the Tone Intercept Treatment, measured in milliseconds. A default of '0' duration is interpreted as the use of a pre-determined duration of tone.
    - Destination Name, Destination Type and Destination Circuit are optional.

- If **Announcement** is chosen then the user should provide
  - Announcement Identifier: is typically a URL of size (0..511).
  - Answer : It can be either YES/NO
  - Barge-In: It can be either YES/NO
  - Destination Name, Destination Type and Destination Circuit are optional
- If **Relcall** is chosen, the treatment is to release call with a cause and if the call is SIP incoming, an optional SIP reason header may also be provisioned. (not applicable to the OpenScape Voice system)

26.7.2.2 Modify an Existing Treatment

To modify an existing treatment, the user is prompted to provide the name of the Intercept and the corresponding parameters.

Caution: The type of the treatment cannot be modified.

- If the treatment type is TONE, the Tone Name, and the duration of the tone may be modified.
- If the treatment type is ANNOUNCE, the announcement message, announcement identifier, the answer parameter of the announcement, the bargeIn parameter of the announcement, and the destination of the announcement can be modified
- If the treatment type is RELCALL, only the Sip Reason Id field may be modified.

26.7.2.3 Unassign a Treatment

To unassign an existing treatment from an intercept, the user is prompted to provide:

- Name of the Intercept,
- Treatment Order.

If there are no remaining treatments assigned to the intercept, by unassigning the treatment, the intercept cycle and interval of the Intercept are reset to default values.

26.7.2.4 Display a Treatment

To display existing treatments for an intercept, the user is prompted to provide:

- Intercept Name
- Treatment Order
- Page Size
- Page Number

If the Intercept Name is given, it acts as a simple filter on the set of Treatments that may be retrieved. The paging parameters are optional, and if not passed, the behavior defaults to returning all existing Treatments with the maximum of 1000 entries. For more information on Paging, refer to Section 5.5, “Paging”, on page 72.

For each entry, the Intercept Name, Cycle, Interval, Type and the corresponding parameters according to the Treatment Type (Tone/Announce) is returned.

Table 957 outlines the interfaces to support the provisioning of treatments.

Return Value	Function
struct unsp__AssignTreatmentResult	int unsp__assignTreatment (unsp__TreatmentData AssignTreatmentRequest, struct unsp__AssignTreatmentResult * AssignTreatmentResult)
struct unsp__ModifyTreatmentResult	int unsp__modifyTreatment (unsp__ModifyTreatmentData ModifyTreatmentRequest, struct unsp__ModifyTreatmentResult * ModifyTreatmentResult)
struct unsp__UnassignTreatmentResult	int unsp__unassignTreatment (unsp__UnassignTreatmentId UnassignTreatmentRequest, struct unsp__UnassignTreatmentResult * UnassignTreatmentResult)
struct unsp__QueryTreatmentDataResult	int unsp__queryTreatment (unsp__QueryTreatmentRequest QueryTreatmentRequest, struct unsp__QueryTreatmentDataResult * QueryTreatmentResult)

Table 957 Treatment Interfaces

26.8 PRIMECLUSTER (PCL) Management

This new V3.1 feature supports the OpenScape Voice system redundancy with geographic separation of the two cluster nodes, which ensures high availability in a catastrophic building failure scenario. Specifically, it will allow the cluster interconnection between the two nodes to be a Layer 3 IP connection. To do this, seventeen (17) new RTP parameters will be added to the RtpParameters.conf

file, under the Small Medium Enterprise (SME) framework. The Soap interface is responsible for displaying these PRIMECLUSTER interconnect configuration parameters Conveniently, they are listed below, but are also added to Table 949.

Note: The SME is an integrated subset of the SOAP server. It has its own framework and generates the library libxdmsubp.so. Most of the commonly used objects for simple Enterprise environment are contained in the library. The SME framework is being used for some objects in the feature even though they are outside of the Enterprise environment.

Srx/Main/ClusterName	Srx/Main/node1CIPLogical
Srx/Main/node1Name	Srx/Main/node2CIP0
Srx/Main/node1PhyIpAddr	Srx/Main/node2CIP1
Srx/Main/node2Name	Srx/Main/node2Logical
Srx/Main/node2PhyIpAddr	Srx/Main/node1CIPNetmask
Srx/Main/ClusterType	Srx/Main/node1CIPNetwork
Srx/Main/node1CIP0	Srx/Main/node2CIPNetmask
Srx/Main/node1CIP1	Srx/Main/node2CIPNetwork
Srx/Main/CIPBroadcast	

27 RTP Parameters Relevant to Subscriber Provisioning

Table 958 lists and describes the RTP Configuration Parameters that are used by the SOAP Server and the Subscriber Management Schedule process during subscriber provisioning activities. The behavior of the SOAP Server depends on the configured values for these parameters. Please refer to Section 7.3, “Get Config Data”, on page 95 for a description of the interface used to retrieve the current values of these parameters.

Parameter	Description
Srx/Main/SvcAssignedPerOfficeACR	Indicates whether the ACR feature is assigned on an office-wide basis.
Srx/Main/CustomerType	Indicates whether the system is configured for use in the Enterprise environment. The soap Processes use this parameter to: • respond to the GetVersion information, • control the services that are available on the system via the FeatureList.dat file
Srx/Xdm/DnCode/digits/minLen	Indicates the minimum allowable directory number length.
Srx/Xdm/DnCode/digits/maxLen	Indicates the maximum allowable directory number length.
Srx/Xdm/DnCode/npa/minLen	Indicates the minimum allowable area code (NPA) length.
Srx/Xdm/DnCode/npa/maxLen	Indicates the maximum allowable area code (NPA) length.
Srx/Xdm/HomeDn/digits/minLen	Indicates the minimum allowable home DN length.
Srx/Xdm/HomeDn/digits/maxLen	Indicates the maximum allowable home DN length.
Srx/Xdm/OfficeCode/digits/minLen	Indicates the minimum allowable office code length.
Srx/Xdm/OfficeCode/digits/maxLen	Indicates the maximum allowable office code length.
Srx/Xdm/OfficeCode/countryCode/minLen	Indicates the minimum allowable country code length.
Srx/Xdm/OfficeCode/countryCode/maxLen	Indicates the maximum allowable country code length.
Srx/Sip/AuthEnabled	Indicates whether SIP Authorization is enabled.
Srx/Sip/max_number_of_sessions	Indicates the maximum allowable number of SIP sessions.
Srx/Sip/default_number_of_sessions	Indicates the default number of SIP sessions.
Srx/Subp/Address	Indicates the IP address on which the SOAP Server should listen to receive SOAP requests. SoapServer will process requests only on this adapter. To accept requests on any adapter specify 0.0.0.0. This parameter defaults to 0.0.0.0.
Srx/Subp/Port	Indicates the port on which the first instance of the SOAP server should listen to receive SOAP requests. When NumberOfInstances > 1, this is the first of NumberOfInstances ports. This parameter defaults to 8767.
Srx/Subp/NumberOfInstances	Indicates the number of SOAP server child processes to run. Each child processes requests on a single port. This parameter defaults to 4.

Table 958 RTP Configuration Parameters (Sheet 1 of 4)

Parameter	Description
Srx/Subp/OverLoadCheckPeriod	Indicates the time period after which to check for overload conditions on the system. Subscriber provisioning checks the congestion level after this time period. It rejects requests for this period of time when in congestion. This parameter defaults to 30 seconds.
Srx/Subp/OverLoadLevel1Sleep	The number of seconds to sleep when the system is in congestion level 1. The default is 30.
Srx/Subp/OverLoadLevel2Sleep	The number of seconds to sleep when the system is in congestion level 2. The default is 60.
Srx/Subp/OverLoadLevel3Sleep	The number of seconds to sleep when the system is in congestion level 3. The default is 120.
Srx/Subp/JobCheckSleep	Indicates the number of seconds for the submgtSchedule process to pause between checks for new jobs. This parameter defaults to 10.
Srx/Subp/NumObjectsToPauseAfter	Indicates the number of objects after which the submgtSchedule process pauses. SubmgtSchedule cascades feature changes to multiple subscribers. This parameter is used to limit the impact of this processing on the switch. This parameter defaults to 20.
Srx/Subp/JobLoopSleepTime	Indicates the number of seconds for the submgtSchedule process to pause between each object processed. This parameter is used to limit the impact of this processing on the switch. This parameter defaults to 0.
Srx/Subp/JobClearTime	Indicates the time of day at which the submgtSchedule process checks for and removes old jobs. The (jobs) results of the feature cascadings are maintained for a user-defined number of days. This parameter reflects the time of day that these files are purged. The format is 24HH:MM. This parameter defaults to 02:00.
Srx/Subp/ClearJobDays	Indicates the number of days after which the submgtSchedule process removes successfully completed jobs. This parameter defaults to 3 days.
Srx/Subp/ClearErrorDays	Indicates the number of days after which the submgtSchedule process removes jobs that ended abnormally. This parameter defaults to 5 days.
Srx/Main/VMWIcapabilityDefault	Indicates whether the Visual Message Waiting Indication capability is Provisioned automatically when the Voice Mail feature is provisioned. This parameter defaults to True.
Srx/Main/InvalidStringChars	Defines a set of characters, which, if included in any character string parameter in a provisioning request, causes the request to be rejected.
Srx/Main/DisableDNValidation	This parameter, if defined and set to RtpTrue, causes the SOAP Server to perform no validation on forward-to numbers. The default is RtpFalse.
Srx/Main/AllowCFDestEmergency	A flag that may be set to allow subscribers to set their Call Forwarding destinations to Emergency Numbers (like 911). The default is RtpFalse.

Table 958 RTP Configuration Parameters (Sheet 2 of 4)

Parameter	Description
Srx/Main/SwitchFPRestrict	If set to RtpTrue, indicates that a switch-wide Feature Profile cannot be assigned to a BGL - only BG-associated FPs may be assigned to BGLs. If set to RtpFalse, a switch-wide FP can be assigned to BGLs. The default is RtpFalse.
Srx/Main/SrxSwitchLangMode	Defines language support on the switch with one of the following possible values: 1 - Single Language 2 - Multi-Language (meaning that subscribers can be provisioned with their language of choice for announcements) The default value is 1.
Srx/Main/SrxDefaultLang	Defines the default language for the switch. This is the language used when a language code is not specified individually for a subscriber or other entity. The default value is 1 (English).
Srx/Main/SrxSupportedLangList	Defines the list of possible supported languages. The parameter is a comma separated string of up to 20 language codes (for example “1,2,3”). The default value is 1 (English), and the numeric values correspond to other languages defined internally.
Srx/Main/SS7Variant	This identifies the nature of the SS7 variant, with the following possible values: 0 - No SS7 support 1 - ANSI 2 - ITU 3 - Japan 6 - China The default value is 1 (ANSI). Note that this parameter is used to for ANSI/ITU (ETSI) checks in the following scenarios: • Validating certain SignalType and SignalProfileFlags values for Create and Modify Signal Profile SOAP requests. • Obtaining the list of features that are valid for a particular environment in response to the GetFeatureList request. • Determining whether a service feature is allowed to be assigned, if it is ANSI or ITU specific. Note that this check is NOT done if the *Srx/Main/CustomerType* parameter is set to Enterprise.
Srx/Main/hiQVariant	Defines the Application Market Domain, with the following possible values: 1 - US 2 - ETSI The default value is 1 (US). Note that this parameter is used to for ANSI/ITU (ETSI) checks in the following scenarios: • Determining whether to allow hex digits in a dialed DN. • Adding the correct prefixes when checking whether numbers are valid call forwarding destinations. • Adding the correct prefixes when checkiing whether numbers are valid hotline/warmline destinations.

Table 958 RTP Configuration Parameters (Sheet 3 of 4)

Parameter	Description
Srx/Main/ApplicationType	Identifies whether the application type for the switch is Fixed Mobile Convergence (FMC) or not. The possible values are: 1 - Non-FMC 2 - FMC The default value is 1 (non-FMC).
Srx/Main/CFVMCompatibility	Indicates whether it shall be possible to assign the Voice Mail feature as well as one or both of the Call Forwarding Variants Call Forwarding Busy (CFB) and Call Forward No Reply (CFDA) to the same subscriber or feature profile. If this parameter is RtpTrue, the assignments are allowed; if RtpFalse, the CF features are mutually exclusive with Voice Mail.
Srx/Subp/Tombstone/KeepRecordsInterval	The number of days to keep tombstone entries in the database. The default value is 30.
Srx/Main/LCSMaxNumConfPorts	The maximum number of conference participants.
Srx/Main/OutGoingCallingPartyNumberType	Should be set to value 0 for E.164 numbers and value 1 for PNP numbers.
Subp/SoapHomeDNReturnCodeRelease	Indicates the different returncodes for provisioning in V2.1 and V3.1.
Srx/Irm/VideoBitRate_Unknown	type: integer, endtype value: [32000-960000] endvalue
Srx/Irm/VideoBitRate_H263	type: integer, endtype value: [32000-960000] endvalue
Srx/Irm/VideoBitRate_H264_Profile_1	type: integer, endtype value: [32000-64000] endvalue
Srx/Irm/VideoBitRate_H264_Profile_1_b	type: integer, endtype value: [32000-128000] endvalue
Srx/Irm/VideoBitRate_H264_Profile_1_1	type: integer, endtype value: [32000-192000] endvalue
Srx/Irm/VideoBitRate_H264_Profile_1_2	type: integer, endtype value: [32000-384000] endvalue
Srx/Irm/VideoBitRate_H264_Profile_1_3	type: integer, endtype value: [32000-768000] endvalue
Srx/Irm/VideoBitRate_H264_Profile_2	type: integer, endtype value: [32000-2000000] endvalue
Srx/Irm/VideoBitRate_H264_Profile_2_1	type: integer, endtype value: [32000-4000000] endvalue
Srx/Irm/VideoBitRate_H264_Profile_2_2	type: integer, endtype value: [32000-4000000] endvalue
Srx/Irm/VideoBitRate_H264_Profile_3	type: integer, endtype value: [32000-10000000] endvalue
Srx/Irm/VideoBitRate_H264_Profile_3_1	type: integer, endtype value: [32000-14000000] endvalue
Srx/Irm/VideoBitRate_H264_Profile_3_2	type: integer, endtype value: [32000-20000000] endvalue
Srx/Irm/VideoBitRate_H264_Profile_4	type: integer, endtype value: [32000-20000000] endvalue
Srx/Irm/VideoBitRate_H264_Profile_4_1	type: integer, endtype value: [32000-50000000] endvalue
Srx/Irm/VideoBitRate_H264_Profile_4_2	type: integer, endtype value: [32000-50000000] endvalue
Srx/Irm/VideoBitRate_H264_Profile_5	type: integer, endtype value: [32000-135000000] endvalue
Srx/Irm/VideoBitRate_H264_Profile_5_1	type: integer, endtype value: [32000-240000000] endvalue

Table 958 RTP Configuration Parameters (Sheet 4 of 4)

Many RTP parameters can be displayed and/or configured via the SOAP Interface. For details, please refer to Section 7.4, “Set Config Parm Data” and Section 7.5, “Get Config Parm Data”.

The deliverable file **RtpParameters.conf** is used by those Get and Set requests to determine if a particular RTP parameter is one of the subset that can be accessed via the SOAP interface. The current version of this file is available on the system in the /unisphere/srx3000/UNSPsubp directory.

28 WSDL and Example SOAP Messages

The interface definitions are provided only as a sample.

28.1 WSDL Schema Definition

The Web Services Description Language (WSDL) describes the message sets and interfaces provided by the OpenScape Voice system for subscriber provisioning. This interface is defined by the SOAP, XML and HTTP standards. WSDL is SOAP's equivalent of an XML schema or CORBA IDL.

28.2 Interfacing to the System using the WSDL

The WSDL is the interface definition. SOAP and XML support and provide for flexibility in the message processing to allow for backwards compatibility. Changes in the WSDL need not require programming changes. With this in mind — especially since there may be multiple Unify SoftSwitches to be provisioned in a common management network — the customer should process without strict WSDL conformance.

The SOAP client should process messages and allow for various tags not to be sent by the OpenScape Voice system. The SOAP client may raise an alarm, but should not abend or core dump. Additionally new elements (attributes/structures) may be returned from the softswitch in new releases. The customer process may raise alarms or log this event, but it should process the rest of the message and ignore these new data fields.

The software and WSDL interface version are obtainable from the softswitch using the GetVersion interface.

28.3 Sample Messages

The following sections present sample messages that the OpenScape Voice system will receive and return. All messages are request/response. There are no unsolicited outbound subscriber provisioning messages from the OpenScape Voice system. Alerts are different and are addressed in a different section of this document.

In the following examples, please note the following about the syntax.

A parameter sent like this:

```
<AccountId></AccountId>
```

...means that the parameter is being provided with no value. For a string parameter, this would mean that an empty string is being sent, and the SOAP Server would process this request as if the operator intends to replace the existing value of the parameter with the empty string. This is not the same as requesting that an existing value remain unchanged for this parameter.

Alternatively, a parameter sent such as this:

```
<BGAttendantNumber xsi:nil=”true”/>
```

...is equivalent to not sending the parameter at all. In this case the SOAP Server performs whatever default action is defined when the parameter is not provided. For a create operation, this typically means that some predefined default value is assigned. For a modify operation, this typically means that the existing value is unchanged.

28.3.1 Get Version Sample Messages

28.3.1.1 Get Version Request

The following SOAP request is used to query the current WSDL version supported by the OpenScape Voice system.

```
POST / HTTP/1.0
Content-Type: text/xml; charset=utf-8
Accept: application/soap+xml, application/dime, multipart/
related, text/*
User-Agent: Axis/1.2RC2
Host: 10.77.14.10:8767
Cache-Control: no-cache
Pragma: no-cache
SOAPAction: “”
Content-Length: 338
<soapenv:Envelope xmlns:soapenv=”http://
schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://
www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/
2001/XMLSchema-instance">
   <soapenv:Body>
      <ns1:GetVersion soapenv:encodingStyle="http://
schemas.xmlsoap.org/soap/encoding/"
xmlns:ns1="urn:siemens-hiq8000"/>
   </soapenv:Body></soapenv:Envelope>
```

28.3.1.2 Get Version Response

The response will resemble the following message:

```
HTTP/1.1 200 OK
```

```
Server: gSOAP/2.7
Content-Type: text/xml; charset=utf-8
Content-Length: 760
Connection: close
<?xml version="1.0" encoding="UTF-8"?>
   <SOAP-ENV:Envelope xmlns:SOAP-ENV="http://
schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENC="http:/
/schemas.xmlsoap.org/soap/encoding/" xmlns:xsi="http://
www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://
www.w3.org/2001/XMLSchema" xmlns:unsp="urn:siemens-
hiq8000">
      <SOAP-ENV:Body>
         <unsp:GetVersionResult>
            <Result>
               <ResultCode>0</ResultCode>
               <ResultMsgNumber>0</ResultMsgNumber>
               <ResultTOY>2006-02-09T14:07:37</ResultTOY>
               <ResultText1>Success</ResultText1>
            </Result>
            <VersionData>
               <Version>10.00.09.01</Version>
               <MarketBuild>10.01.01.ALL.22</MarketBuild>
               <SoftSwitchEnvironment>Carrier</
SoftSwitchEnvironment>
            <buildDateTime>2006-01-23T15:06</buildDateTime>
            </VersionData>
         </unsp:GetVersionResult>
      </SOAP-ENV:Body>
   </SOAP-ENV:Envelope>
```

28.3.2 Create Subscriber Sample Messages

28.3.2.1 Create Subscriber Request

The following is an example of a SOAP request used to create a subscriber:

```
POST / HTTP/1.0
Content-Type: text/xml; charset=utf-8
Accept: application/soap+xml, application/dime, multipart/
related, text/*
User-Agent: Axis/1.2RC2
Host: 10.77.14.10:8767
Cache-Control: no-cache
Pragma: no-cache
SOAPAction: ""
Content-Length: 3891
<soapenv:Envelope xmlns:soapenv="http://
schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://
www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/
2001/XMLSchema-instance">
   <soapenv:Body>
      <ns1:CreateSubscriber soapenv:encodingStyle="http://
schemas.xmlsoap.org/soap/encoding/"
xmlns:ns1="urn:siemens-hiq8000">
         <ServiceId>5619556559</ServiceId>
         <OperatorId>SuperAdmin</OperatorId>
         <CreateSubRequest>
```

```
                    <AccountMgtInfo>
                       <AccountId></AccountId>
                       <BillingId></BillingId>
                       <RetailerId></RetailerId>
                       <ClassOfService></ClassOfService>
                       <RateArea></RateArea>
                       <TestLineFlag xsi:nil="true"/>
                       <ExternalDNFlag xsi:nil="true"/>
                       <CountryCodeLen>0</CountryCodeLen>
                       <CallingLoc></CallingLoc>
                       <FeatureProfileId>
                          <FeatureProfileName></FeatureProfileName>
                           <BGName xsi:nil="true"/>
                       </FeatureProfileId>
                       <NumberPlanName></NumberPlanName>
                       <ExternalCallerID></ExternalCallerID>
                       <UseMainPilotDN></UseMainPilotDN>
                       <ImpactLevel></ImpactLevel>
                    </AccountMgtInfo>
                    <SubscriberBGInfo>
                       <BGName></BGName>
                       <BGLineName></BGLineName>
                       <BGAttendantNumber xsi:nil="true"/>
                       <CpuId>0</CpuId>
                       <Extension></Extension>
                       <BGDeptName xsi:nil="true"/>
                    </SubscriberBGInfo>
                    <AccountUserInfo>
                       <DisplayName></DisplayName>
                       <TimeZone></TimeZone>
                       <Pin></Pin>
                       <PublicPin></PublicPin>
                       <Language>System Default</Language>
                    </AccountUserInfo>
                    <Status>
                       <SubscriberStatus>Active</SubscriberStatus>
                       <InterceptAnnouncement xsi:nil="true"/>
                       <NewServiceId xsi:nil="true"/>
                    </Status>
                    <ConnectionInfo>
                       <IPPhone>
                          <Name>5619556559</Name>
                          <RegType>Dynamic</RegType>
                          <IPAddr xsi:nil="true"/>
                          <Port xsi:nil="true"/>
                          <Register xsi:nil="true"/>
                          <TimeToLive xsi:nil="true"/>
                          <MaxSessions xsi:nil="true"/>
                          <SIPSecurity>
                             <Realm xsi:nil="true"/>
                             <UserName xsi:nil="true"/>
                             <Password xsi:nil="true"/>
                            <Scheme>digest-authentication</Scheme>
                          </SIPSecurity>
                          <IPConnectionType>UDP</IPConnectionType>
                        <Attributes>
                          <TrustedSub>false</TrustedSub>
                          <GSMSub>false</GSMSub>
                        <  MediaReDirectionAllowed>false</
MediaReDirectionAllowed>
                          <StaticIMSSub>false</StaticIMSSub>
                          <AoRPerContact>false</AoRPerContact>
```

```
                      <OverrideIRMCodecRestrictions>false</
OverrideIRMCodecRestrictions>
                   <TransferHandoff>false</TransferHandoff>
                  <AcceptBillingNumber>false</
AcceptBillingNumber>
                      <SendInsecureRefferedByHeader>false</
SendInsecureRefferedByHeader>
               </Attributes>
                   <ContactInfoList xsi:nil="true"/>
                <AssociatedSipEndpointName xsi:nil="true"/>
               </IPPhone>
               <MGCPPhone xsi:nil="true"/>
               <H323Term xsi:nil="true"/>
            </ConnectionInfo>
            <VOIP>
               <Bearer-Speech>true</Bearer-Speech>
               <Bearer-3-1K xsi:nil="true"/>
               <Bearer-56K xsi:nil="true"/>
               <Bearer-64K xsi:nil="true"/>
               <FaxMode>G711</FaxMode>
               <EchoCancellation>true</EchoCancellation>
             <SilenceSuppression>false</SilenceSuppression>
               <FaxStatistics>false</FaxStatistics>
               <OutbandDTMFSignaling>false</
OutbandDTMFSignaling>
               <SuppressOutbandDTMFWithG711>false</
SuppressOutbandDTMFWithG711>
            </VOIP>
            <QualityOfService xsi:nil="true"/>
            <InterExchange>
               <PIC1></PIC1>
               <PIC2></PIC2>
               <PIC3></PIC3>
               <PIC4></PIC4>
            </InterExchange>
            <KeysetInfo>
               <KeysetUse xsi:nil="true"/>
            </KeysetInfo>
            <ListOfFeatures>
               <CfAC xsi:nil="true"/>
               <CfACR xsi:nil="true"/>
               <CfAR xsi:nil="true"/>
               <CfACCB xsi:nil="true"/>
               <CfAcctCode xsi:nil="true"/>
               <CfAuthCode xsi:nil="true"/>
               <CfBgCNAM xsi:nil="true"/>
               <CfBgCPU xsi:nil="true"/>
               <CfBgCT xsi:nil="true"/>
               <CfBgDAPPS xsi:nil="true"/>
               <CfBgDNPPS xsi:nil="true"/>
               <CfBLV xsi:nil="true"/>
               <CfCCW xsi:nil="true"/>
               <CfCFB xsi:nil="true"/>
               <CfCFDA xsi:nil="true"/>
               <CfCFV xsi:nil="true"/>
               <CfCHLD xsi:nil="true"/>
               <CfCIDCW xsi:nil="true"/>
               <CfCIDS xsi:nil="true"/>
               <CfCISNAME xsi:nil="true"/>
               <CfCISNUM xsi:nil="true"/>
               <CfCNAB xsi:nil="true"/>
               <CfCNAM xsi:nil="true"/>
```

```
                <CfCND xsi:nil="true"/>
                <CfCNDB xsi:nil="true"/>
                <CfCPK xsi:nil="true"/>
                <CfCT xsi:nil="true"/>
                <CfCSTA xsi:nil="true"/>
                <CfLONR xsi:nil="true"/>
                <CfLINR xsi:nil="true"/>
                <CfCCBS xsi:nil="true"/>
                <CfCCNR xsi:nil="true"/>
                <CfCOLR xsi:nil="true"/>
                <CfCW xsi:nil="true"/>
                <CfDAPPS xsi:nil="true"/>
                <CfDNPPS xsi:nil="true"/>
                <CfEACR xsi:nil="true"/>
                <CfECF xsi:nil="true"/>
                <CfHotDesk xsi:nil="true"/>
                <CfIN xsi:nil="true"/>
                <CfMCT xsi:nil="true"/>
                <CfMOH xsi:nil="true"/>
                <CfONS xsi:nil="true"/>
                <CfPRES xsi:nil="true"/>
                <CfRACF xsi:nil="true"/>
                <CfRCF xsi:nil="true"/>
                <CfRFA xsi:nil="true"/>
                <CfSCA xsi:nil="true"/>
                <CfSCF xsi:nil="true"/>
                <CfSCR xsi:nil="true"/>
                <CfSERRNG xsi:nil="true"/>
                <CfSILMProtection xsi:nil="true"/>
                <CfSILM xsi:nil="true"/>
                <CfSPCALL xsi:nil="true"/>
                <CfSR xsi:nil="true"/>
                <CfSRS xsi:nil="true"/>
                <CfTRS xsi:nil="true"/>
                <CfVM xsi:nil="true"/>
             </ListOfFeatures>
          </CreateSubRequest>
       </ns1:CreateSubscriber>
    </soapenv:Body></soapenv:Envelope>
```

28.3.2.2 Create Subscriber Response

The response will resemble the following message:

```
HTTP/1.1 200 OK
Server: gSOAP/2.7
Content-Type: text/xml; charset=utf-8
Content-Length: 624
Connection: close
<?xml version="1.0" encoding="UTF-8"?>
   <SOAP-ENV:Envelope xmlns:SOAP-ENV="http://
schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENC="http:/
/schemas.xmlsoap.org/soap/encoding/" xmlns:xsi="http://
www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://
www.w3.org/2001/XMLSchema" xmlns:unsp="urn:siemens-
hiq8000">
      <SOAP-ENV:Body>
         <unsp:ResultCodeStruct>
            <ResultCode>0</ResultCode>
```

```
            <ResultMsgNumber>0</ResultMsgNumber>
            <ResultTOY>2006-02-09T14:08:55</ResultTOY>
            <ResultText1>Success</ResultText1>
            <ResultText2></ResultText2>
            <ResultText3></ResultText3>
            <ResultText4></ResultText4>
         </unsp:ResultCodeStruct>
      </SOAP-ENV:Body>
   </SOAP-ENV:Envelope>
```

28.3.3 Get Subscriber Info Sample Messages

28.3.3.1 Get Subscriber Info Request

The following message will retrieve the data for the specified subscriber:

```
POST / HTTP/1.0
Content-Type: text/xml; charset=utf-8
Accept: application/soap+xml, application/dime, multipart/
related, text/*
User-Agent: Axis/1.2RC2
Host: 10.77.14.10:8767
Cache-Control: no-cache
Pragma: no-cache
SOAPAction: ""
Content-Length: 570
<soapenv:Envelope xmlns:soapenv="http://
schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://
www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/
2001/XMLSchema-instance">
   <soapenv:Body>
      <ns1:GetSubInfoByRel soapenv:encodingStyle="http://
schemas.xmlsoap.org/soap/encoding/"
xmlns:ns1="urn:siemens-hiq8000">
         <GetSubInfoRequest>
            <InterfaceRelease xsi:nil="true"/>
            <InterfaceWSDL xsi:nil="true"/>
            <ServiceId>5619556559</ServiceId>
            <GetSubDataOptions>
               <GetSubscriberDataOptionList/>
            </GetSubDataOptions>
         </GetSubInfoRequest>
      </ns1:GetSubInfoByRel>
   </soapenv:Body></soapenv:Envelope>
```

28.3.3.2 Get Subscriber Info Response

The response will resemble the following message:

```
HTTP/1.1 200 OK
Server: gSOAP
<ResultCode>0
<ResultText1>Success
<ResultText2>
```

```
<ResultText3>
<ResultText4>
<?xml>


  <SOAP-ENV:Envelope>
    <SOAP-ENV:Body>
      <unsp:GetSubscriberResult>
        <Result>
          <ResultCode>0</ResultCode>
          <ResultMsgNumber>0</ResultMsgNumber>
          <ResultTOY>2008-07-01T10:30:10</ResultTOY>
          <ResultText1>Success</ResultText1>
          <ResultText2></ResultText2>
          <ResultText3></ResultText3>
          <ResultText4></ResultText4>
        </Result>
        <ServiceId>5619551110</ServiceId>
        <ServiceIdType>PrimaryDn</ServiceIdType>
        <PrimaryDnInfo>
          <OfficeCodeInfo>
            <OfficeCode>561955</OfficeCode>
            <CountryCodeLength>0</CountryCodeLength>
            <AreaCodeLength>3</AreaCodeLength>
            <Attributes>AtrNo</Attributes>
          </OfficeCodeInfo>
          <AccountMgtInfo>
            <AccountId></AccountId>
            <BillingId>5619551110</BillingId>
            <RetailerId></RetailerId>
            <ClassOfService></ClassOfService>
            <RateArea></RateArea>
            <TestLineFlag>false</TestLineFlag>
            <ExternalDNFlag>false</ExternalDNFlag>
            <CountryCodeLen>0</CountryCodeLen>
            <CallingLoc></CallingLoc>
            <FeatureProfileId>
              <FeatureProfileName></FeatureProfileName>
              <BGName></BGName>
            </FeatureProfileId>
            <NumberPlanName>E164NANP</NumberPlanName>
            <ExternalCallerID></ExternalCallerID>
            <UseMainPilotDN></UseMainPilotDN>
            <ImpactLevel></ImpactLevel>
          </AccountMgtInfo>
          <AccountUserInfo>
            <DisplayName></DisplayName>
            <UnicodeDisplayName></UnicodeDisplayName>
            <TimeZone>LOCAL</TimeZone>
            <Pin></Pin>
            <PublicPin></PublicPin>
            <Language>System Default</Language>
          </AccountUserInfo>
          <Status>
            <SubscriberStatus>Active</SubscriberStatus>
            <InterceptAnnouncement>None</
InterceptAnnouncement>
            <NewServiceId></NewServiceId>
          </Status>
          <ConnectionInfo>
            <IPPhone>
```

```
                <Name>5619551110</Name>
                <RegType>Dynamic</RegType>
                <Port>5060</Port>
                <LataDigits></LataDigits>
                <Register>false</Register>
                <TimeToLive>-1</TimeToLive>
                <MaxSessions>0</MaxSessions>
                <IPConnectionType>UDP</IPConnectionType>
                <Attributes>
                  <TrustedSub>false</TrustedSub>
                  <GSMSub>false</GSMSub>
                  <MediaReDirectionAllowed>false</
MediaReDirectionAllowed>
                  <StaticIMSSub>false</StaticIMSSub>
                  <AoRPerContact>false</AoRPerContact>
                  <AcceptBillingNumber>false</
AcceptBillingNumber>
                  <SendInsecureRefferedByHeader>false</
SendInsecureRefferedByHeader>
                  <TrustedSub>false</TrustedSub>
                  <GSMSub>false</GSMSub>
                  <MediaReDirectionAllowed>false</
MediaReDirectionAllowed>
                  <StaticIMSSub>false</StaticIMSSub>
                  <AoRPerContact>false</AoRPerContact>
                    <OverrideIRMCodecRestrictions>false</
OverrideIRMCodecRestrictions>
                  <TransferHandoff>false</TransferHandoff>
                  <AcceptBillingNumber>false</
AcceptBillingNumber>
                    <SendInsecureRefferedByHeader>false</
SendInsecureRefferedByHeader>
                </Attributes>
                <AssociatedSipEndpointName></
AssociatedSipEndpointName>
                <LooseRoutingURI></LooseRoutingURI>
<SubscriberLicenseType>UndefinedSubscriberLicense</
SubscriberLicenseType>
              <SIPSecurity>
                <SrtpSupport>SrtpDisabled</SrtpSupport>
              </SIPSecurity>
              </IPPhone>
            </ConnectionInfo>
            <VOIP>
              <Bearer-Speech>true</Bearer-Speech>
              <Bearer-3-1K>false</Bearer-3-1K>
              <Bearer-56K>false</Bearer-56K>
              <Bearer-64K>false</Bearer-64K>
              <FaxMode>G711</FaxMode>
              <EchoCancellation>false</EchoCancellation>
              <SilenceSuppression>false</SilenceSuppression>
              <FaxStatistics>false</FaxStatistics>
              <OutbandDTMFSignaling>false</
OutbandDTMFSignaling>
              <SuppressOutbandDTMFWithG711>false</
SuppressOutbandDTMFWithG711>
            </VOIP>
            <QualityOfService>
              <QOSClass>
                <QOSClassMode>TOS</QOSClassMode>
                <QOSValue>0</QOSValue>
              </QOSClass>
```

```
            <QOSPrimary>
              <Codec>None</Codec>
              <PacketizationPeriod>0</PacketizationPeriod>
              <TypeOfService>0</TypeOfService>
              <ResourceReservation>BestEffort</
ResourceReservation>
            </QOSPrimary>
            <QOSSecondary>
              <Codec>None</Codec>
              <PacketizationPeriod>0</PacketizationPeriod>
              <TypeOfService>0</TypeOfService>
              <ResourceReservation>BestEffort</
ResourceReservation>
            </QOSSecondary>
            <QOSTertiary>
              <Codec>None</Codec>
              <PacketizationPeriod>0</PacketizationPeriod>
              <TypeOfService>0</TypeOfService>
              <ResourceReservation>BestEffort</
ResourceReservation>
            </QOSTertiary>
            <QOSFourth>
              <Codec>None</Codec>
              <PacketizationPeriod>0</PacketizationPeriod>
              <TypeOfService>0</TypeOfService>
              <ResourceReservation>BestEffort</
ResourceReservation>
            </QOSFourth>
          </QualityOfService>
          <InterExchange>
            <PIC1></PIC1>
            <PIC2></PIC2>
            <PIC3></PIC3>
            <PIC4></PIC4>
          </InterExchange>
          <KeysetInfo>
            <KeysetUse>KeysetNone</KeysetUse>
          </KeysetInfo>
          <SubscriberAuditInfo>
            <InitialProvTOY>2008-07-01T10:30:03</
InitialProvTOY>
            <LastOpProvTOY>2008-07-01T10:30:03</
LastOpProvTOY>
            <LastOpId>as</LastOpId>
          </SubscriberAuditInfo>
          <ListOfFeatures>
            <CfMCT>
              <Subscribed>SwitchWide</Subscribed>
            </CfMCT>
          </ListOfFeatures>
          <SubMlhgInfo>
            <MlhgOwner>false</MlhgOwner>
          </SubMlhgInfo>
        </PrimaryDnInfo>
      </unsp:GetSubscriberResult>
    </SOAP-ENV:Body>
  </SOAP-ENV:Envelope>
```

28.3.4 Update Subscriber Features Sample Messages

28.3.4.1 Update Subscriber Features Request

The following SOAP request will assign the Call Forwarding Busy (CFB) service feature to a subscriber:

```
POST / HTTP/1.0
Content-Type: text/xml; charset=utf-8
Accept: application/soap+xml, application/dime, multipart/
related, text/*
User-Agent: Axis/1.2RC2
Host: 10.77.14.10:8767
Cache-Control: no-cache
Pragma: no-cache
SOAPAction: ""
Content-Length: 2282
<soapenv:Envelope xmlns:soapenv="http://
schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://
www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/
2001/XMLSchema-instance">
   <soapenv:Body>
      <ns1:UpdateSubscriberFeatures
soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/
encoding/" xmlns:ns1="urn:siemens-hiq8000">
         <ServiceId>5619556559</ServiceId>
         <OperatorId>SuperAdmin</OperatorId>
         <ListOfFeatures>
            <CfAC xsi:nil="true"/>
            <CfACR xsi:nil="true"/>
            <CfAR xsi:nil="true"/>
            <CfACCB xsi:nil="true"/>
            <CfAcctCode xsi:nil="true"/>
            <CfAuthCode xsi:nil="true"/>
            <CfBgCNAM xsi:nil="true"/>
            <CfBgCPU xsi:nil="true"/>
            <CfBgCT xsi:nil="true"/>
            <CfBgDAPPS xsi:nil="true"/>
            <CfBgDNPPS xsi:nil="true"/>
            <CfBLV xsi:nil="true"/>
            <CfCCW xsi:nil="true"/>
            <CfCFB>
               <Subscribed>AddReplace</Subscribed>
               <SubActivatable>None</SubActivatable>
               <DestServiceIdSubControllable>None</
DestServiceIdSubControllable>
               <CFCourtesyCallBehavior>NoCourtesyCall</
CFCourtesyCallBehavior>
               <Enabled>false</Enabled>
               <ServiceId></ServiceId>
             <CFNotifyCallingParty>CfNotifyCallingPtyNone</
CFNotifyCallingParty>
               <NotifySubCallFwdActive>false</
NotifySubCallFwdActive>
               <Local xsi:nil="true"/>
            </CfCFB>
            <CfCFDA xsi:nil="true"/>
            <CfCFV xsi:nil="true"/>
```

```
                <CfCHLD xsi:nil="true"/>
                <CfCIDCW xsi:nil="true"/>
                <CfCIDS xsi:nil="true"/>
                <CfCISNAME xsi:nil="true"/>
                <CfCISNUM xsi:nil="true"/>
                <CfCNAB xsi:nil="true"/>
                <CfCNAM xsi:nil="true"/>
                <CfCND xsi:nil="true"/>
                <CfCNDB xsi:nil="true"/>
                <CfCPK xsi:nil="true"/>
                <CfCT xsi:nil="true"/>
                <CfCSTA xsi:nil="true"/>
                <CfLONR xsi:nil="true"/>
                <CfLINR xsi:nil="true"/>
                <CfCCBS xsi:nil="true"/>
                <CfCCNR xsi:nil="true"/>
                <CfCOLR xsi:nil="true"/>
                <CfCW xsi:nil="true"/>
                <CfDAPPS xsi:nil="true"/>
                <CfDNPPS xsi:nil="true"/>
                <CfEACR xsi:nil="true"/>
                <CfECF xsi:nil="true"/>
                <CfHotDesk xsi:nil="true"/>
                <CfIN xsi:nil="true"/>
                <CfMCT xsi:nil="true"/>
                <CfMOH xsi:nil="true"/>
                <CfONS xsi:nil="true"/>
                <CfPRES xsi:nil="true"/>
                <CfRACF xsi:nil="true"/>
                <CfRCF xsi:nil="true"/>
                <CfRFA xsi:nil="true"/>
                <CfSCA xsi:nil="true"/>
                <CfSCF xsi:nil="true"/>
                <CfSCR xsi:nil="true"/>
                <CfSERRNG xsi:nil="true"/>
                <CfSILMA xsi:nil="true"/>
                <CfSILMS xsi:nil="true"/>
                <CfSPCALL xsi:nil="true"/>
                <CfSR xsi:nil="true"/>
                <CfSRS xsi:nil="true"/>
                <CfTRS xsi:nil="true"/>
                <CfTWC xsi:nil="true"/>
                <CfVM xsi:nil="true"/>
            </ListOfFeatures>
        </ns1:UpdateSubscriberFeatures>
    </soapenv:Body></soapenv:Envelope>
```

28.3.4.2 Update Subscriber Features Response

The response will resemble the following message:

```
HTTP/1.1 200 OK
Server: gSOAP/2.7
Content-Type: text/xml; charset=utf-8
Content-Length: 624
Connection: close
<?xml version="1.0" encoding="UTF-8"?>
   <SOAP-ENV:Envelope xmlns:SOAP-ENV="http://
schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENC="http:/
```

```
/schemas.xmlsoap.org/soap/encoding/" xmlns:xsi="http://
www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://
www.w3.org/2001/XMLSchema" xmlns:unsp="urn:siemens-
hiq8000">
      <SOAP-ENV:Body>
         <unsp:ResultCodeStruct>
            <ResultCode>0</ResultCode>
            <ResultMsgNumber>0</ResultMsgNumber>
            <ResultTOY>2006-02-09T14:28:13</ResultTOY>
            <ResultText1>Success</ResultText1>
            <ResultText2></ResultText2>
            <ResultText3></ResultText3>
            <ResultText4></ResultText4>
         </unsp:ResultCodeStruct>
      </SOAP-ENV:Body>
   </SOAP-ENV:Envelope>
```

28.3.5 Update Subscriber PIC Sample Messages

28.3.5.1 Update Subscriber PIC Request

The following message will change the subscriber's interlata and international carrier (PIC1):

```
POST / HTTP/1.0
Content-Type: text/xml; charset=utf-8
Accept: application/soap+xml, application/dime, multipart/
related, text/*
User-Agent: Axis/1.2RC2
Host: 10.77.14.10:8767
Cache-Control: no-cache
Pragma: no-cache
SOAPAction: ""
Content-Length: 529
<soapenv:Envelope xmlns:soapenv="http://
schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://
www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/
2001/XMLSchema-instance">
   <soapenv:Body>
      <ns1:UpdateSubscriberPICs
soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/
encoding/" xmlns:ns1="urn:siemens-hiq8000">
         <ServiceId>5619556559</ServiceId>
         <OperatorId>SuperAdmin</OperatorId>
         <InterExchange>
            <PIC1>0110</PIC1>
            <PIC2></PIC2>
            <PIC3></PIC3>
            <PIC4></PIC4>
         </InterExchange>
      </ns1:UpdateSubscriberPICs>
   </soapenv:Body></soapenv:Envelope>
```

28.3.5.2 Update Subscriber PIC Response

The response will resemble the following message:

```
HTTP/1.1 200 OK
Server: gSOAP/2.7
Content-Type: text/xml; charset=utf-8
Content-Length: 624
Connection: close
<?xml version="1.0" encoding="UTF-8"?>
   <SOAP-ENV:Envelope xmlns:SOAP-ENV="http://
schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENC="http:/
/schemas.xmlsoap.org/soap/encoding/" xmlns:xsi="http://
www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://
www.w3.org/2001/XMLSchema" xmlns:unsp="urn:siemens-
hiq8000">
      <SOAP-ENV:Body>
         <unsp:ResultCodeStruct>
            <ResultCode>0</ResultCode>
            <ResultMsgNumber>0</ResultMsgNumber>
            <ResultTOY>2006-02-09T14:45:12</ResultTOY>
            <ResultText1>Success</ResultText1>
            <ResultText2></ResultText2>
            <ResultText3></ResultText3>
            <ResultText4></ResultText4>
         </unsp:ResultCodeStruct>
      </SOAP-ENV:Body></SOAP-ENV:Envelope>
```

28.3.6 Update Subscriber Account Management Info Sample Messages

28.3.6.1 Update Subscriber Account Management Info Request

The following message will change the subscriber's Account ID and Billing ID:

```
POST / HTTP/1.0
Content-Type: text/xml; charset=utf-8
Accept: application/soap+xml, application/dime, multipart/
related, text/*
User-Agent: Axis/1.2RC2
Host: 10.77.14.10:8767
Cache-Control: no-cache
Pragma: no-cache
SOAPAction: ""
Content-Length: 857
<soapenv:Envelope xmlns:soapenv="http://
schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://
www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/
2001/XMLSchema-instance">
   <soapenv:Body>
      <ns1:UpdateSubscriberAccountMgtInfo
soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/
encoding/" xmlns:ns1="urn:siemens-hiq8000">
```

```
            <ServiceId>5619556559</ServiceId>
            <OperatorId>SuperAdmin</OperatorId>
            <AccountMgtInfo>
               <AccountId>account6559</AccountId>
               <BillingId>5619556559</BillingId>
               <RetailerId></RetailerId>
               <ClassOfService></ClassOfService>
               <RateArea></RateArea>
               <TestLineFlag xsi:nil="true"/>
               <ExternalDNFlag xsi:nil="true"/>
               <CountryCodeLen xsi:nil="true"/>
               <CallingLoc></CallingLoc>
               <FeatureProfileId>
                  <FeatureProfileName xsi:nil="true"/>
                  <BGName xsi:nil="true"/>
               </FeatureProfileId>
               <NumberPlanName></NumberPlanName>
               <ExternalCallerID></ExternalCallerID>
               <UseMainPilotDN></UseMainPilotDN>
               <ImpactLevel></ImpactLevel>
            </AccountMgtInfo>
         </ns1:UpdateSubscriberAccountMgtInfo>
      </soapenv:Body></soapenv:Envelope>
```

28.3.6.2 Update Subscriber Account Management Info Response

The response will resemble the following message:

```
HTTP/1.1 200 OK
Server: gSOAP/2.7
Content-Type: text/xml; charset=utf-8
Content-Length: 624
Connection: close
<?xml version="1.0" encoding="UTF-8"?>
   <SOAP-ENV:Envelope xmlns:SOAP-ENV="http://
schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENC="http:/
/schemas.xmlsoap.org/soap/encoding/" xmlns:xsi="http://
www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://
www.w3.org/2001/XMLSchema" xmlns:unsp="urn:siemens-
hiq8000">
      <SOAP-ENV:Body>
         <unsp:ResultCodeStruct>
            <ResultCode>0</ResultCode>
            <ResultMsgNumber>0</ResultMsgNumber>
            <ResultTOY>2006-02-20T14:09:37</ResultTOY>
            <ResultText1>Success</ResultText1>
            <ResultText2></ResultText2>
            <ResultText3></ResultText3>
            <ResultText4></ResultText4>
         </unsp:ResultCodeStruct>
      </SOAP-ENV:Body>
   </SOAP-ENV:Envelope>
```

28.3.7 Update Subscriber Account User Info Sample Messages

28.3.7.1 Update Subscriber Account User Info Request

The following message will change the subscriber's PIN:

```
POST / HTTP/1.0
Content-Type: text/xml; charset=utf-8
Accept: application/soap+xml, application/dime, multipart/
related, text/*
User-Agent: Axis/1.2RC2
Host: 10.77.14.10:8767
Cache-Control: no-cache
Pragma: no-cache
SOAPAction: ""
Content-Length: 625
<soapenv:Envelope xmlns:soapenv="http://
schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://
www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/
2001/XMLSchema-instance">
   <soapenv:Body>
      <ns1:UpdateSubscriberAccountUserInfo
soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/
encoding/" xmlns:ns1="urn:siemens-hiq8000">
         <ServiceId>5619556559</ServiceId>
         <OperatorId>SuperAdmin</OperatorId>
         <AccountUserInfo>
            <DisplayName></DisplayName>
            <TimeZone>LOCAL</TimeZone>
            <Pin>5432</Pin>
            <PublicPin></PublicPin>
            <Language>System Default</Language>
         </AccountUserInfo>
      </ns1:UpdateSubscriberAccountUserInfo>
   </soapenv:Body></soapenv:Envelope>
```

28.3.7.2 Update Subscriber Account User Info Response

The response will resemble the following message:

```
HTTP/1.1 200 OK
Server: gSOAP/2.7
Content-Type: text/xml; charset=utf-8
Content-Length: 624
Connection: close
<?xml version="1.0" encoding="UTF-8"?>
   <SOAP-ENV:Envelope xmlns:SOAP-ENV="http://
schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENC="http:/
/schemas.xmlsoap.org/soap/encoding/" xmlns:xsi="http://
www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://
www.w3.org/2001/XMLSchema" xmlns:unsp="urn:siemens-
hiq8000">
      <SOAP-ENV:Body>
```

```
         <unsp:ResultCodeStruct>
            <ResultCode>0</ResultCode>
            <ResultMsgNumber>0</ResultMsgNumber>
            <ResultTOY>2006-02-20T14:12:30</ResultTOY>
            <ResultText1>Success</ResultText1>
            <ResultText2></ResultText2>
            <ResultText3></ResultText3>
            <ResultText4></ResultText4>
         </unsp:ResultCodeStruct>
      </SOAP-ENV:Body>
   </SOAP-ENV:Envelope>
```

28.3.8 Update Subscriber Status Sample Messages

28.3.8.1 Update Subscriber Status Request

The following message will change the subscriber's status to Blocked:

```
POST / HTTP/1.0
Content-Type: text/xml; charset=utf-8
Accept: application/soap+xml, application/dime, multipart/
related, text/*
User-Agent: Axis/1.2RC2
Host: 10.77.14.10:8767
Cache-Control: no-cache
Pragma: no-cache
SOAPAction: ""
Content-Length: 490
<soapenv:Envelope xmlns:soapenv="http://
schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://
www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/
2001/XMLSchema-instance">
   <soapenv:Body>
      <ns1:UpdateSubscriberStatus
soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/
encoding/" xmlns:ns1="urn:siemens-hiq8000">
         <ServiceId>5619556559</ServiceId>
         <OperatorId>SuperAdmin</OperatorId>
         <SubscriberStatus>Blocked</SubscriberStatus>
      </ns1:UpdateSubscriberStatus>
   </soapenv:Body></soapenv:Envelope>
```

28.3.8.2 Update Subscriber Status Response

The response will resemble the following message:

```
HTTP/1.1 200 OK
Server: gSOAP/2.7
Content-Type: text/xml; charset=utf-8
Content-Length: 624
Connection: close
<?xml version="1.0" encoding="UTF-8"?>
   <SOAP-ENV:Envelope xmlns:SOAP-ENV="http://
schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENC="http:/
```

```
/schemas.xmlsoap.org/soap/encoding/" xmlns:xsi="http://
www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://
www.w3.org/2001/XMLSchema" xmlns:unsp="urn:siemens-
hiq8000">
      <SOAP-ENV:Body>
         <unsp:ResultCodeStruct>
            <ResultCode>0</ResultCode>
            <ResultMsgNumber>0</ResultMsgNumber>
            <ResultTOY>2006-02-20T14:14:48</ResultTOY>
            <ResultText1>Success</ResultText1>
            <ResultText2></ResultText2>
            <ResultText3></ResultText3>
            <ResultText4></ResultText4>
         </unsp:ResultCodeStruct>
      </SOAP-ENV:Body>
   </SOAP-ENV:Envelope>
```

28.3.9 Update Subscriber Connection Info Sample Messages

28.3.9.1 Update Subscriber Connection Info Request

The following message will change the subscriber's Max Sessions to 6:

```
POST / HTTP/1.0
Content-Type: text/xml; charset=utf-8
Accept: application/soap+xml, application/dime, multipart/
related, text/*
User-Agent: Axis/1.2RC2
Host: 10.77.14.10:8767
Cache-Control: no-cache
Pragma: no-cache
SOAPAction: ""
Content-Length: 959
<soapenv:Envelope xmlns:soapenv="http://
schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://
www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/
2001/XMLSchema-instance">
   <soapenv:Body>
      <ns1:UpdateConnectionInfo
soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/
encoding/" xmlns:ns1="urn:siemens-hiq8000">
         <ServiceId>5619556559</ServiceId>
         <OperatorId>SuperAdmin</OperatorId>
         <ConnectionUpdateRequest>
            <IPPhoneUpdateInfo>
               <RegType>Dynamic</RegType>
               <IPAddr xsi:nil="true"/>
               <Port>5060</Port>
               <Register xsi:nil="true"/>
               <TimeToLive xsi:nil="true"/>
               <MaxSessions>6</MaxSessions>
               <SIPSecurity>
                  <Realm></Realm>
                  <UserName></UserName>
                  <Password></Password>
```

```
                  <Scheme>digest-authentication</Scheme>
               <SrtpSupport>SrtpEnabled</SrtpSupport>
               </SIPSecurity>
               <IPConnectionType>UDP</IPConnectionType>
               <Attributes>
                  <TrustedSub>false</TrustedSub>
                  <GSMSub>false</GSMSub>
                  <MediaReDirectionAllowed>false</
MediaReDirectionAllowed>
                     <StaticIMSSub>false</StaticIMSSub>
                  <AoRPerContact>false</AoRPerContact>
                      <OverrideIRMCodecRestrictions>false<
/OverrideIRMCodecRestrictions>
                  <TransferHandoff>false</TransferHandoff>
                  <AcceptBillingNumber>false</
AcceptBillingNumber>
                  <SendInsecureRefferedByHeader>false</
SendInsecureRefferedByHeader>
            </Attributes>
               <AssociatedSipEndpointName xsi:nil="true"/>
            </IPPhoneUpdateInfo>
            <MGCPPhoneUpdateInfo xsi:nil="true"/>
            <H323TermUpdateInfo xsi:nil="true"/>
         </ConnectionUpdateRequest>
      </ns1:UpdateConnectionInfo>
   </soapenv:Body></soapenv:Envelope>
```

28.3.9.2 Update Subscriber Connection Info Response

The response will resemble the following message:

```
HTTP/1.1 200 OK
Server: gSOAP/2.7
Content-Type: text/xml; charset=utf-8
Content-Length: 624
Connection: close
<?xml version="1.0" encoding="UTF-8"?>
   <SOAP-ENV:Envelope xmlns:SOAP-ENV="http://
schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENC="http:/
/schemas.xmlsoap.org/soap/encoding/" xmlns:xsi="http://
www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://
www.w3.org/2001/XMLSchema" xmlns:unsp="urn:siemens-
hiq8000">
      <SOAP-ENV:Body>
         <unsp:ResultCodeStruct>
            <ResultCode>0</ResultCode>
            <ResultMsgNumber>0</ResultMsgNumber>
            <ResultTOY>2006-02-20T14:18:29</ResultTOY>
            <ResultText1>Success</ResultText1>
            <ResultText2></ResultText2>
            <ResultText3></ResultText3>
            <ResultText4></ResultText4>
         </unsp:ResultCodeStruct>
      </SOAP-ENV:Body>
   </SOAP-ENV:Envelope>
```

28.3.10 Update Subscriber DN Sample Messages

28.3.10.1 Update Subscriber DN Request

The following message will change the subscriber's DN.

```
POST / HTTP/1.0
Content-Type: text/xml; charset=utf-8
Accept: application/soap+xml, application/dime, multipart/
related, text/*
User-Agent: Axis/1.2RC2
Host: 10.77.14.10:8767
Cache-Control: no-cache
Pragma: no-cache
SOAPAction: ""
Content-Length: 571
<soapenv:Envelope xmlns:soapenv="http://
schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://
www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/
2001/XMLSchema-instance">
   <soapenv:Body>
      <ns1:UpdateSubscriberDn soapenv:encodingStyle="http:/
/schemas.xmlsoap.org/soap/encoding/"
xmlns:ns1="urn:siemens-hiq8000">
         <ServiceId>5619556559</ServiceId>
         <OperatorId>SuperAdmin</OperatorId>
         <DnChangeRequest>
            <InterceptAnnouncement>NumberChange</
InterceptAnnouncement>
            <NewServiceId>5619556568</NewServiceId>
         </DnChangeRequest>
      </ns1:UpdateSubscriberDn>
   </soapenv:Body></soapenv:Envelope>
```

28.3.10.2 Update Subscriber DN Response

The response will resemble the following message:

```
HTTP/1.1 200 OK
Server: gSOAP/2.7
Content-Type: text/xml; charset=utf-8
Content-Length: 624
Connection: close
<?xml version="1.0" encoding="UTF-8"?>
   <SOAP-ENV:Envelope xmlns:SOAP-ENV="http://
schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENC="http:/
/schemas.xmlsoap.org/soap/encoding/" xmlns:xsi="http://
www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://
www.w3.org/2001/XMLSchema" xmlns:unsp="urn:siemens-
hiq8000">
      <SOAP-ENV:Body>
         <unsp:ResultCodeStruct>
            <ResultCode>0</ResultCode>
            <ResultMsgNumber>0</ResultMsgNumber>
            <ResultTOY>2006-02-20T14:58:08</ResultTOY>
```

```
            <ResultText1>Success</ResultText1>
            <ResultText2></ResultText2>
            <ResultText3></ResultText3>
            <ResultText4></ResultText4>
         </unsp:ResultCodeStruct>
      </SOAP-ENV:Body>
   </SOAP-ENV:Envelope>
```

28.3.10.3 Delete Subscriber Request

The following message will delete the specified subscriber:

```
POST / HTTP/1.0
Content-Type: text/xml; charset=utf-8
Accept: application/soap+xml, application/dime, multipart/
related, text/*
User-Agent: Axis/1.2RC2
Host: 10.77.14.10:8767
Cache-Control: no-cache
Pragma: no-cache
SOAPAction: ""
Content-Length: 434
<soapenv:Envelope xmlns:soapenv="http://
schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://
www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/
2001/XMLSchema-instance">
   <soapenv:Body>
      <ns1:DeleteSubscriber soapenv:encodingStyle="http://
schemas.xmlsoap.org/soap/encoding/"
xmlns:ns1="urn:siemens-hiq8000">
         <ServiceId>5619556559</ServiceId>
         <OperatorId>SuperAdmin</OperatorId>
      </ns1:DeleteSubscriber>
   </soapenv:Body></soapenv:Envelope>
```

28.3.10.4 Delete Subscriber Response

The response will resemble the following message:

```
HTTP/1.1 200 OK
Server: gSOAP/2.7
Content-Type: text/xml; charset=utf-8
Content-Length: 624
Connection: close
<?xml version="1.0" encoding="UTF-8"?>
   <SOAP-ENV:Envelope xmlns:SOAP-ENV="http://
schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-ENC="http:/
/schemas.xmlsoap.org/soap/encoding/" xmlns:xsi="http://
www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://
www.w3.org/2001/XMLSchema" xmlns:unsp="urn:siemens-
hiq8000">
      <SOAP-ENV:Body>
         <unsp:ResultCodeStruct>
            <ResultCode>0</ResultCode>
            <ResultMsgNumber>0</ResultMsgNumber>
            <ResultTOY>2006-02-20T13:55:59</ResultTOY>
```

```
            <ResultText1>Success</ResultText1>
            <ResultText2></ResultText2>
            <ResultText3></ResultText3>
            <ResultText4></ResultText4>
         </unsp:ResultCodeStruct>
      </SOAP-ENV:Body>
   </SOAP-ENV:Envelope>
```

28.3.11 Get COS List Sample Messages

28.3.11.1 Get COS List Request

The following message will retrieve the list of Class Of Service (COS) values defined on the system:

```
POST / HTTP/1.0
Content-Type: text/xml; charset=utf-8
Accept: application/soap+xml, application/dime, multipart/
related, text/*
User-Agent: Axis/1.2RC2
Host: 10.77.14.10:8767
Cache-Control: no-cache
Pragma: no-cache
SOAPAction: ""
Content-Length: 529
<soapenv:Envelope xmlns:soapenv="http://
schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://
www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/
2001/XMLSchema-instance">
<SOAP-ENV:Body>
<unsp:GetCOSList>
</unsp:GetCOSList>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

28.3.11.2 Get COS List Response

The response will resemble the following message:

```
HTTP/1.1 200 OK
Server: gSOAP
<ResultCode>0
<ResultText1>Success
<ResultText2>
<ResultText3>
<ResultText4>
<?xml>
  <SOAP-ENV:Envelope>
    <SOAP-ENV:Body>
      <unsp:GetCOSListResult>
        <Result>
          <ResultCode>0</ResultCode>
```

```
            <ResultMsgNumber>0</ResultMsgNumber>
            <ResultTOY>2006-02-09T14:46:47</ResultTOY>
            <ResultText1>Success</ResultText1>
            <ResultText2></ResultText2>
            <ResultText3></ResultText3>
            <ResultText4></ResultText4>
          </Result>
          <GetCOSListData>
            <COS>FGDIN</COS>
            <COS>Ordinary</COS>
<COS>Prison</COS>
            <COS>Hotel</COS>
            <COS>Hospital</COS>
            <COS>OptrScreenReq</COS>
            <COS>OpenScapeApp</COS>
          </GetCOSListData>
        </unsp:GetCOSListResult>
      </SOAP-ENV:Body>
    </SOAP-ENV:Envelope>
```

29 FeatureList.dat

The Unify platform software is delivered into a number of different solutions (OpenScape Voice, HiPath 4200, HiPath 4000). The features supported on these solutions can differ. The processing of these features by the SOAP Server is controlled using a combination of the Node.cfg (RTP parms) and a flat file called FeatureList.dat.

The FeatureList.dat file is used to validate supported features (when assigning features) and in the GetFeatureList (Advertising features available on the system). The OpenScape Voice Assistant uses the GetFeatureList to determine what features to display on the screens.

Below outlines and defines the format and processing of this file.

The file is a comma delimited, keyword associated file, with the following syntax for each line:

FeatureName,FeatureNumericId,”Feature Description”condition1=boolean, Condition2=bool

where:

- FeatureName is the WSDL supported featurename (CfAR,CfCW...)
- FeatureNumericId is a unique number for use by the OpenScape Voice Assistant.
- Feature Description is the name of the feature.
- Conditions = <*See Table 959 for appropriate setting for each keyword.*>

Example:

Keyword	Setting
sub	is true if the feature is assignable to a residential subscriber
bg	is true if the feature is assignable to a business group
bgl	is true if the feature is assignable to a business group subscriber
ep	is true if the feature is assignable to an endpoint profile
fp_centrex	is true if the feature is assignable to a centrex feature profile
fp_residential	is true if the feature is assignable to a residential feature profile
sip	is true if the feature is assignable to a subscriber with a SIP phone
mgcp	is true if the feature is assignable to a subscriber that is connected via MGCP. Not for OpenScape Voice.
ansi	is true when the feature is assignable in the US market
etsi	is true when the feature is assignable in the world market
enterprise	is true if the feature is assignable in the enterprise environment

Table 959 FeatureList.dat Keyword and Settings (Sheet 1 of 2)

Keyword	Setting
mlhg_pilot	is true if the feature is assignable to the pilot of a mlhg.

Table 959 FeatureList.dat Keyword and Settings (Sheet 2 of 2)

CfCND,11,"Incoming Number Delivery", sub=1, bg=0, bgl=1, ep=0, fp_centrex=1, fp_residential=1, sip=1, mgcp=1, ansi=1, etsi=1, noss7=1, carrier=1, enterprise=1, mlhg_pilot=1, teen=0

When assigning a feature to an entity, all applicable flags for that entity and the environment are checked. All applicable flags must be set to true for the feature to be applied.

If the environment is Enterprise, the Ansi / Etsi flags do not apply.

For information about the RTP parameters that are used to determine the solution and environment (Carrier/Enterprise and ANSI/ETSI in particular), refer to Chapter 27, “RTP Parameters Relevant to Subscriber Provisioning”.

A Time Zones

The subscriber may belong to a specific time zone. This time zone defines how the subscriber's time-dependent services work. In general, most subscribers will be in the system's local time zone, but SIP phones can be anywhere and the system can border a time zone edge and serve multiple time zones.

If no time zone is specified, the subscriber is treated as being in the system’s local time zone.

The possible time zone values are as follows.

Africa/Abidjan
Africa/Accra
Africa/Addis_Ababa
Africa/Algiers
Africa/Asmera
Africa/Bamako
Africa/Bangui
Africa/Banjul
Africa/Bissau
Africa/Blantyre
Africa/Brazzaville
Africa/Bujumbura
Africa/Cairo
Africa/Casablanca
Africa/Ceuta
Africa/Conakry
Africa/Dakar
Africa/Dar_es_Salaam
Africa/Djibouti
Africa/Douala
Africa/El_Aaiun
Africa/Freetown
Africa/Gaborone
Africa/Harare
Africa/Johannesburg
Africa/Kampala
Africa/Khartoum
Africa/Kigali
Africa/Kinshasa
America/Vancouver
America/Virgin
America/Whitehorse
America/Winnipeg
America/Yakutat
America/Yellowknife
Antarctica/Casey
Antarctica/Davis
Antarctica/DumontDUrville
Antarctica/Mawson
Antarctica/McMurdo
Antarctica/Palmer
Antarctica/South_Pole
Antarctica/Syowa
Antarctica/Vostok
Arctic/Longyearbyen
Asia/Aden
Asia/Almaty
Asia/Amman
Asia/Anadyr
Asia/Aqtau
Asia/Aqtobe
Asia/Ashgabat
Asia/Ashkhabad
Asia/Baghdad
Asia/Bahrain
Asia/Baku
Asia/Bangkok
Asia/Beirut
Europe/Copenhagen
Europe/Dublin
Europe/Gibraltar
Europe/Helsinki
Europe/Istanbul
Europe/Kaliningrad
Europe/Kiev
Europe/Lisbon
Europe/Ljubljana
Europe/London
Europe/Luxembourg
Europe/Madrid
Europe/Malta
Europe/Minsk
Europe/Monaco
Europe/Moscow
Europe/Nicosia
Europe/Oslo
Europe/Paris
Europe/Prague
Europe/Riga
Europe/Rome
Europe/Samara
Europe/San_Marino
Europe/Sarajevo
Europe/Simferopol
Europe/Skopje
Europe/Sofia
Europe/Stockholm

Africa/Lagos
Africa/Libreville
Africa/Lome
Africa/Luanda
Africa/Lubumbashi
Africa/Lusaka
Africa/Malabo
Africa/Maputo
Africa/Maseru
Africa/Mbabane
Africa/Mogadishu
Africa/Monrovia
Africa/Nairobi
Africa/Ndjamena
Africa/Niamey
Africa/Nouakchott
Africa/Ouagadougou
Africa/Porto-Novo
Africa/Sao_Tome
Africa/Timbuktu
Africa/Tripoli
Africa/Tunis
Africa/Windhoek
America/Adak
America/Anchorage
America/Anguilla
America/Antigua
America/Araguaina
America/Aruba
America/Asuncion
America/Atka
America/Barbados
America/Belem
America/Belize
America/Boa_Vista
America/Bogota
America/Boise
America/Buenos_Aires
America/Cambridge_Bay
Asia/Bishkek
Asia/Brunei
Asia/Calcutta
Asia/Choibalsan
Asia/Chongqing
Asia/Chungking
Asia/Colombo
Asia/Dacca
Asia/Damascus
Asia/Dhaka
Asia/Dili
Asia/Dubai
Asia/Dushanbe
Asia/Gaza
Asia/Harbin
Asia/Hong_Kong
Asia/Hovd
Asia/Irkutsk
Asia/Istanbul
Asia/Jakarta
Asia/Jayapura
Asia/Jerusalem
Asia/Kabul
Asia/Kamchatka
Asia/Karachi
Asia/Kashgar
Asia/Katmandu
Asia/Krasnoyarsk
Asia/Kuala_Lumpur
Asia/Kuching
Asia/Kuwait
Asia/Macao
Asia/Macau
Asia/Magadan
Asia/Makassar
Asia/Manila
Asia/Muscat
Asia/Nicosia
Asia/Novosibirsk
Europe/Tallinn
Europe/Tirane
Europe/Tiraspol
Europe/Uzhgorod
Europe/Vaduz
Europe/Vatican
Europe/Vienna
Europe/Vilnius
Europe/Warsaw
Europe/Zagreb
Europe/Zaporozhye
Europe/Zurich
GB
GB-Eire
GMT
GMT+0
GMT+1
GMT+10
GMT+11
GMT+12
GMT+13
GMT+2
GMT+3
GMT+4
GMT+5
GMT+6
GMT+7
GMT+8
GMT+9
GMT-0
GMT-1
GMT-10
GMT-11
GMT-12
GMT-2
GMT-3
GMT-4
GMT-5
GMT-6

America/Cancun
America/Caracas
America/Catamarca
America/Cayenne
America/Cayman
America/Chicago
America/Chihuahua
America/Cordoba
America/Costa_Rica
America/Cuiaba
America/Curacao
America/Danmarkshavn
America/Dawson
America/Dawson_Creek
America/Denver
America/Detroit
America/Dominica
America/Edmonton
America/Eirunepe
America/El_Salvador
America/Ensenada
America/Fort_Wayne
America/Fortaleza
America/Glace_Bay
America/Godthab
America/Goose_Bay
America/Grand_Turk
America/Grenada
America/Guadeloupe
America/Guatemala
America/Guayaquil
America/Guyana
America/Halifax
America/Havana
America/Hermosillo
America/Indiana/Indianapolis
America/Indiana/Knox
America/Indiana/Marengo
America/Indiana/Vevay

Asia/Omsk
Asia/Oral
Asia/Phnom_Penh
Asia/Pontianak
Asia/Pyongyang
Asia/Qatar
Asia/Qyzylorda
Asia/Rangoon
Asia/Riyadh
Asia/Riyadh87
Asia/Riyadh88
Asia/Riyadh89
Asia/Saigon
Asia/Sakhalin
Asia/Samarkand
Asia/Seoul
Asia/Shanghai
Asia/Singapore
Asia/Taipei
Asia/Tashkent
Asia/Tbilisi
Asia/Tehran
Asia/Tel_Aviv
Asia/Thimbu
Asia/Thimphu
Asia/Tokyo
Asia/Ujung_Pandang
Asia/Ulaanbaatar
Asia/Ulan_Bator
Asia/Urumqi
Asia/Vientiane
Asia/Vladivostok
Asia/Yakutsk
Asia/Yekaterinburg
Asia/Yerevan
Atlantic/Azores
Atlantic/Bermuda
Atlantic/Canary
Atlantic/Cape_Verde

GMT-7
GMT-8
GMT-9
GMT0
Greenwich
Hongkong
HST
Iceland
Indian/Antananarivo
Indian/Chagos
Indian/Christmas
Indian/Cocos
Indian/Comoro
Indian/Kerguelen
Indian/Mahe
Indian/Maldives
Indian/Mauritius
Indian/Mayotte
Indian/Reunion
Iran
Israel
Jamaica
Japan
Kwajalein
Libya
MET
Mexico/BajaNorte
Mexico/BajaSur
Mexico/General
Mideast/Riyadh87
Mideast/Riyadh88
Mideast/Riyadh89
MST
MST7MDT
Navajo
NZ
NZ-CHAT
Pacific/Apia
Pacific/Auckland

America/Indianapolis
America/Inuvik
America/Iqaluit
America/Jamaica
America/Jujuy
America/Juneau
America/Kentucky/Louisville
America/Kentucky/Monticello
America/Knox_IN
America/La_Paz
America/Lima
America/Los_Angeles
America/Louisville
America/Maceio
America/Managua
America/Manaus
America/Martinique
America/Mazatlan
America/Mendoza
America/Menominee
America/Merida
America/Mexico_City
America/Miquelon
America/Monterrey
America/Montevideo
America/Montreal
America/Montserrat
America/Nassau
America/New_York
America/Nipigon
America/Nome
America/Noronha
America/North_Dakota/Center
America/Panama
America/Pangnirtung
America/Paramaribo
America/Phoenix
America/Port-au-Prince
America/Port_of_Spain

Atlantic/Faeroe
Atlantic/Jan_Mayen
Atlantic/Madeira
Atlantic/Reykjavik
Atlantic/South_Georgia
Atlantic/St_Helena
Atlantic/Stanley
Australia/ACT
Australia/Adelaide
Australia/Brisbane
Australia/Broken_Hill
Australia/Canberra
Australia/Darwin
Australia/Broken_Hill
Australia/Canberra
Australia/Hobart
Australia/LHI
Australia/Lindeman
Australia/Lord_Howe
Australia/Melbourne
Australia/North
Australia/NSW
Australia/Perth
Australia/Queensland
Australia/South
Australia/Sydney
Australia/Tasmania
Australia/Victoria
Australia/West
Australia/Yancowinna
Brazil/Acre
Brazil/DeNoronha
Brazil/East
Brazil/West
Canada/Atlantic
Canada/Central
Canada/East-Saskatchewan
Canada/Eastern
Canada/Mountain

Pacific/Chatham
Pacific/Easter
Pacific/Efate
Pacific/Enderbury
Pacific/Fakaofo
Pacific/Fiji
Pacific/Funafuti
Pacific/Galapagos
Pacific/Gambier
Pacific/Guadalcanal
Pacific/Guam
Pacific/Honolulu
Pacific/Johnston
Pacific/Kiritimati
Pacific/Kosrae
Pacific/Kwajalein
Pacific/Majuro
Pacific/Marquesas
Pacific/Midway
Pacific/Nauru
Pacific/Niue
Pacific/Norfolk
Pacific/Noumea
Pacific/Pago_Pago
Pacific/Palau
Pacific/Pitcairn
Pacific/Ponape
Pacific/Port_Moresby
Pacific/Rarotonga
Pacific/Saipan
Pacific/Samoa
Pacific/Tahiti
Pacific/Tarawa
Pacific/Tongatapu
Pacific/Truk
Pacific/Wake
Pacific/Wallis
Pacific/Yap
Poland

America/Porto_Acre
America/Porto_Velho
America/Puerto_Rico
America/Rainy_River
America/Rankin_Inlet
America/Recife
America/Regina
America/Rio_Branco
America/Rosario
America/Santiago
America/Santo_Domingo
America/Sao_Paulo
America/Scoresbysund
America/Shiprock
America/St_Johns
America/St_Kitts
America/St_Lucia
America/St_Thomas
America/St_Vincent
America/Swift_Current
America/Tegucigalpa
America/Thule
America/Thunder_Bay
America/Tijuana
America/Tortola
Canada/Newfoundland
Canada/Pacific
Canada/Saskatchewan
Canada/Yukon
CET
Chile/Continental
Chile/EasterIsland
CST6CDT
Cuba
EET
Egypt
Eire
EST
EST5EDT
Europe/Amsterdam
Europe/Andorra
Europe/Athens
Europe/Belfast
Europe/Belgrade
Europe/Berlin
Europe/Bratislava
Europe/Brussels
Europe/Bucharest
Europe/Budapest
Europe/Chisinau
Portugal
PRC
PST8PDT
ROK
Singapore
Turkey
UCT
Universal
US/Alaska
US/Aleutian
US/Arizona
US/Central
US/East-Indiana
US/Eastern
US/Hawaii
US/Indiana-Starke
US/Michigan
US/Mountain
US/Pacific
US/Pacific-New
US/Samoa
UTC
W-SU
WET
Zulu

B Test Tools

The following command line test tools are delivered onto the system.

B.1 TestSendSoapReq

This utility parses an input file, sends the request to a soap server, and prints out the resultCode structure or dumps out the xml response.

Parameters:

```
[-h=Host] -p=port -f=xmlfile [-t=testcasename] [-trace]
```

If host is not specified, localhost (127.0.0.1) is defaulted to.

Port is the port number on which the SOAP server is listening on.

Xml file is a file that contains one or more xml messages. Each xml message is encapsulated with the **`<TEST name>`** and **`</TEST name>`** tags.

Testcasename is the testcase from the file to send. If omitted, all testcases are sent.

Trace indicates that the response should be printed out.

Sample xmlfile:

// comments can occur anytime before or after testcases (but not in the middle of).

The host and length information in the http header portion are ignored.

```
<TEST GetVersion>
POST /cgi-bin/CL_Server HTTP/1.0
Host: 10.76.1.80
Content-Type: text/xml; charset=utf-8
Content-Length: 4406
SOAPAction: ""

<?xml version="1.0" encoding="UTF-8"?>

<SOAP-ENV:Envelope

   xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/
envelope/"

   xmlns:SOAP-ENC="http://schemas.xmlsoap.org/soap/
encoding/"

   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

   xmlns:xsd="http://www.w3.org/2001/XMLSchema"
```

```
   xmlns:unsp="urn:siemens-hiq8000"

  SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/
encoding/">

  <SOAP-ENV:Body>

<unsp:GetVersion>

</unsp: GetVersion >

  </SOAP-ENV:Body>

</SOAP-ENV:Envelope>

</TEST GetVersion >
```

B.2 TestSoap.ksh (soapCli)

This utility prompts the operator (similar to the CLI) to issue different SOAP provisioning requests on and to the local soap server. It can be used to exercise all of the different WSDL interfaces.

C Performance

The soapServer will process requests on multiple ports. The numbers defined in this section apply to requests on one port only and do not take into account the overlap improvement of requests being distributed to multiple ports.

Multiple instances of SoapMassProv may also be executed simultaneously. The numbers defined in this section apply to only a single instance running. No performance numbers have been run against soapMassProv. These times will be the same of better than the times of soapServer.

The tests were performed on a single node development machine (Sun 1400 Netra) that is equivalently configured (same performance) as in a customer environment - however there was no call load on the system.

Jmeter was used to send repetitive requests and obtain timings, although full tracing of the soapServer was enabled and a sampling of the times was verified using the log file data. Table 960 shows the average amount of time the soapServer takes to completely process a particular request. For example, creating a SIP subscriber takes an average of 181 milliseconds.

Subscriber Request	Subscriber Type	Time
Create Subscriber • No features • No BG Info		
	SIP	181 ms
Get Subscriber Info • One feature (CNAM) • No BG Info		
	SIP	103 ms
Delete Subscriber • One feature (CNAM) • No BG Info		
	SIP	287 ms
Update Subscriber Features	Add Single Feature (CNAM)	46 ms
	Remove Single Feature (CNAM)	47 ms

Table 960 Average Amount of Time to Process Requests

C.1 Priority Processing and Congestion Controls

The subscriber provisioning processes operate at a lower priority that most other processes on the system. This is so that the impact to call traffic is minimized. The priority is defined in an RTP configuration parameter as defined in Chapter 27, “RTP Parameters Relevant to Subscriber Provisioning”.

The subscriber provisioning processes also participate in congestion control in the event that the system becomes overloaded. The OpenScape Voice system has three congestion levels dependent on CPU load; typically the thresholds are:

- Minor (Level 1): 90% CPU load
- Major (Level 2): 95% CPU load
- Critical (Level 3) 98% CPU load

When the resources of the system become highly used, subscriber provisioning will start rejecting provisioning requests. Table 961 defines the requests and the congestion level at which they are rejected by the SOAP Server. Commands not included in this list are not rejected; for example, GetVersion, which is expected to have negligible impact on processor usage, does not check for congestion before handling the request. The requests listed below are rejected if the congestion level is the indicated level or higher.

The import and export processes soapMassProv and soapExport internally call the same common functions that are called when SOAP requests are handled by the SOAP Server. Therefore, they are subject to the same behavior as listed above, where applicable. For example, if the congestion level is 1, a CSM command in a soapMassProv command file would be rejected because it calls the CreateSubscriber function. The export of subscribers using soapExport would fail if the congestion level is 2 or higher, since it internally calls GetSubscriberList.

The Subscriber Management Scheduling process (can be referred to as the Job Scheduler) also takes steps to avoid interfering with higher-priority system operations in case of overload conditions. The submgtSchedule process checks congestion levels periodically during its operation. By default, the check period is 30 seconds and is configurable via the Srx/Subp/OverLoadCheckPeriod RTP configuration parameter. If an overload condition is detected, the submgtSchedule process suspends processing of the current job and sleeps for 30, 60, or 120 seconds, depending on whether the congestion level is 1, 2, or 3 respectively.

Provisioning Request	Congestion Level
GetSubscriberInfo (and derivatives)	2
GetSubscriberList	2
DeleteSubscriber	1
DisconnectSubscriber	2

Table 961 Requests and Congestion Levels(Sheet 1 of 4)

Provisioning Request	Congestion Level
UpdateSubscriberAccountInfo	2
UpdateSubscriberAccountMgtInfo	2
UpdateSubscriberCapabilities	2
UpdateSubscriberDn	2
UpdateSubscriberFeatures	2
UpdateSubscriberPIC	2
UpdateSubscriberQOS	2
UpdateSubscriberStatus	2
UpdateConnectionInfo	2
CreateMlhg	1
ModifyMlhgInfo	1
DeleteMlhg	1
GetMlhgInfo	2
AddSubToMlhg	2
ModifyMlhgTermInfo	2
DeleteMlhgTerm	2
GetMlhgTMData	2
CreateEpProfile	1
GetEpProfile	2
GetEpList	2
DeleteEpProfile	1
UpdateEpCapabilities	2
UpdateEpFeatures	2
UpdateEpMgtInfo	2
UpdateEpPICs	2
UpdateEpQos	2
CreateBG	1
GetGBInfo	2
GetGBList	2
DeleteBG	1
UpdateBGFeatures	2
UpdateBGParms	2
AddBGMainNumber	2
DeleteBGMainNumber	2

Table 961 Requests and Congestion Levels(Sheet 2 of 4)

Provisioning Request	Congestion Level
UpdateBGMainNumber	2
GetBGAttendantNumbers	2
UpdateKeysetInfo	2
GetKeysetPrimaryList	2
CreatePrefixAccessCode	1
GetPrefixAccessCode	2
DeletePrefixAccessCode	1
ModifyPrefixAccessCode	2
GetDestServiceList	2
CreatePNPLocationCode	1
GetPNPLocationCode	2
DeletePNPLocationCode	1
ModifyPNPLocationCode	2
CreatePNPExtension	1
GetPNPExtension	2
DeletePNPExtension	1
ModifyPNPExtension	2
CreateDestCode	1
GetDestCode	2
DeleteDestCode	1
ModifyDestCode	2
GetDestinationList	2
CreateNumberPlan	1
GetNumberPlan	2
DeleteNumberPlan	1
CreateBgSubnet	1
GetBgSubnetInfo	2
DeleteBgSubnet	1
UpdateBgSubnet	2
GetBgSubnetList	2

Table 961 Requests and Congestion Levels(Sheet 3 of 4)

Provisioning Request	Congestion Level
ModifyBgDept	1
GetBGDeptList	1
CreateAuthCode	1
GetAuthCode	1
DeleteAuthCode	1
GetAuthCodeList	2
AddContactList	1
GetContactList	2
DeleteContactList	2
CreateQOSProfile	1
GetQOSProfile	2
DeleteQOSProfile	1
UpdateQOSProfile	2
GetQOSProfileList	2
CreateFeatureProfile	1
GetFeatureProfile	2
DeleteFeatureProfile	1
UpdateFeatureProfile	2
GetFeatureProfileList	1
CreateGw	1
GetGwInfo	2
DeleteGw	1
UpdateGw	1
GetGwList	2
GetAssociatedSipEndpointList	2
GetInterceptList	2
GetSupportedLanguageList	1
GetTombStoneSubscriberList	2

Table 961 Requests and Congestion Levels(Sheet 4 of 4)

Index

A
access, CLI 31
account code 374
AddBGMainNumber 431
AddContactList 367
AddSubToMlhg 637
alarms 750
anonymous caller rejection 375
attendant answering position 375
attendant numbers 53
authentication methods 58
AUTH_EP 59
authorization code 377

B
background jobs 658
BG only features
    911 LIN administration 394
    attendant answering position 375
    BG distinctive ringing 396
    keyset operation 399
    silent monitoring 417
    traffic measurement 417
BG parking lots 479
BG subnet E911 LIN administration 53
branch offices 488, 492, 504
business group services 52
business groups
    adding main number 431
    authorization codes 465
    concepts 52
    creating 429
    departments 53, 443
    feature dependencies 36
    parking lots 479
    subnet E911 LIN administration 53
    subnets 447
    traffic measurements 54
    updating 430

C
call forward busy 384
call forward LED indication 57
call forward no reply 386
Call Forward System Internal External 379
call forward unconditional 382
call forward unreachable 382
call forwarding restrictions 419
call forwarding voice mail 388
call pickup 390
called party name delivery 391
called party number delivery 391
calling name delivery (BG) 391
calling name delivery block 391
calling number delivery 392
calling number delivery block 392
CallingLocation
    get list 100, 101
CancelJob 661
CfAcctCode – account code 374
CfACR – anonymous caller rejection 375
CfAuthCode – authorization code 377
CfBgCNAM – calling name delivery (BG) 391
CfBgCPU – call pickup 390
CfBgDR – BG distinctive ringing 396
CfBgE911LIN – E911 LIN administration 394
CfBgHotDesk – hot desking (BG) 399
CfBgKS – keyset operation 399
CfBgNBCPU – night bell CPU 375
CfBgSILM – silent monitoring 417
CfBgSPCALL – speed call (speed dial) (BG) 412
CfBgSPCALL – station speed call (BG) 411
CfBgTM – traffic measurement 417
CfCCBS – completion of calls on busy subscriber 378
CfCCNR – Completion of Calls on No Reply 378
CfCFB – call forward busy 384
CfCFD – call forward dependable 382
CfCFDA – call forward no reply 386
CfCFID – call forward invalid 379
CfCFR – call forwarding restrictions 419
CfCFSIE – call forward system internal external 379
CfCFV – call forward unconditional 382
CfCIDS – outgoing caller ID suppression 391
CfCISNAME – called party name delivery 391
CfCISNUM – called party number delivery 391
CfCNAB – calling name delivery block 391
CfCNAM – incoming call name delivery 391
CfCND – calling number delivery 392
CfCNDB – calling number delivery block 392
CfCPS – call park to server 420
CfCSTA – CSTA for SimplyPhone 392
CfDND – do not disturb 397
CfDTF – Deny Terminating Feature 393
CfECF – enhanced call forward 395, 413

CfHotDesk – hot desking 398
CfMCT – customer originated trace 393
CfMOH – music on hold 401
CfONS – one number service 402
CfRACF – remote activation call forward 403
CfRCF – remote call forward 404
CfSCA – selective caller accept 405
CfSCF – selective call forward 406
CfSCR – selective caller reject 407
CfSERRNG – serial ringing 408
CfSILM – silent monitoring 418
CfSILMProtection – silent monitoring protection 418
CfSILMTag – silent monitoring tag 418
CfSR – station restriction (BG) 410
CfSRS – simultaneous ringing 409
CfTRS – toll restrictions 414
CfVM – call forwarding voice mail 388
CLI
    See command line interface.
command line interface 31
Common Community Groups Return Codes 505
completion of calls on busy subscriber 378
Completion of Calls on No Reply 378
COS
    get list 99
Create Class Of Restrictions 708
Create DNM Conversion 597
Create DNM Defintion 569
Create DNM LocalToll 605
Create DNM Modification 579
Create DNM Normalization 589
Create DNM Prefix 560
Create DNM Prefix Input Fields 561
Create Subscriber
    Language 325
    PublicPin 325
create subscriber
    language 312
    PublicPin 312
create subscriber sample messages
    create subscriber request 915
    create subscriber response 918
Create Traffic Type 703, 733
CreateAuthCode 465
CreateBG 429
CreateDestCode 523
CreateDestination 153, 172
CreateEpProfile 644
CreateFeatureProfile 613
CreateMlhg 622
CreateMlhgTerm 640
CreatePnpExtension 554
CreatePrefixAccessCode 535
CreateRateArea 159
CreateSubscriber 303
creating
    BG authorization codes 465
    business groups 429
    destination codes 523
    hunt group terminals 640
    multiple hunt groups 622
    PNP extension 554
    prefix access codes 535
CRM
    See customer relationship management.
CSTA for SimplyPhone 392
customer originated trace 393
customer relationship management
    interface version support 33
    required features 32
    required interface support 33
    standards support 33

D

Delete Class Of Restrictions 710
Delete DNM Conversion 601
Delete DNM Definition 572
Delete DNM LocalToll 609
Delete DNM Modification 583
Delete DNM Normalization 593
Delete DNM Prefix 563
delete subscriber sample messages
    delete subscriber request 933
    delete subscriber response 933
Delete Traffic Type 706, 747
DeleteAuthCode 466
DeleteBGDept 445
DeleteBgSubnet 457
DeleteContactList 368
DeleteDestCode 530
DeleteDestination 156, 176
DeleteEndpoint 151
DeleteFeatureProfile 615
DeleteMlhg 632
DeletePrefixAccessCode 545
DeleteSubscriber 345
deleting
    BG authorization code 466
    BG departments 445
    BG subnet 457
    destination codes 530
    multiple hunt groups 632
    PNP location codes 552
    prefix access codes 545

departments 53
design considerations
CLI 31
CRM 32
security 31
sizing 31
system requirements 31
traffic volumes 31
destination codes 521
DisconnectSubscriber 347
display number definition table 47
display number modification table 50
display number prefix table 48
DisplayDestination 157, 176, 177, 178
DisplayEndpoint 151
Do Not Disturb 397

E

E.164 number plan 55
E911 LIN administration 53
end point profile 644
enhanced call forward 395, 413
error removing gold jobs alarm data 753

F

feature profiles 34
Features
call forward invalid 379
call forward system internal external 379
completion of calls on no reply 378
CSTA for SimplyPhone 392
enhanced call forward 395
One Number Service 402
Out Callers Prem Presentation 397
features
account code 374
anonymous caller rejection 375
authorization code 377
business groups 375
call forward busy 384
call forward no reply 386
call forward unconditional 382
call forwardig voice mail 388
call forwarding restrictions 419
call pickup 390
called party name delivery 391
called party number delivery 391
calling name delivery (BG) 391
calling name delivery block 391
calling number delivery 392
calling number delivery block 392
completion of calls on busy subscriber 378
CRM 32
customer originated trace 393
enhanced call forward 413
hot desking 398
hot desking (BG) 399
incoming call name delivery 391
keyset operation 57
music on hold 401
outgoing caller ID suppression 391
remote activation call forward 403
remote call forward 404
selective call forward 406
selective caller accept 405
selective caller reject 407
serial ringing 408
silent monitoring 418
silent monitoring protection 418
silent monitoring tag 418
simultaneous ringing 409
station restrictions (BG) 410
station speed call (BG) 411
toll restrictions 414
field data type 66
field definitions 67

G

general utility interfaces 94
get COS list sample messages
get COS list request 934
get COS list response 934
Get Local CPU Members List 504
get version sample messages
get version request 914
get version response 914
GetBGAttendantNumbers 440
GetBGInfo 436
GetBGInfoByOptions 436
GetBGList 437
GetConfigData 95
GetContactList 369
GetDestCodeList 532
GetDestServiceList 534
GetEpInfo 653
GetEpInfoByRelease 655
GetEPList 655
GetFeatureList 94
GetFeatureProfile 617
GetFeatureProfileList 616
GetJobInfo 658
GetJobList 660
GetJobListByOption 660
GetKeysetPrimaryList 344

GetMlhgTMData 643
GetNumberPlanIDInfo 521
GetNumberPlanList 519
GetPrefixAccessCodeList 546
GetPrefixAccessCodeOne 549
GetSubInfoByRel 333
GetSubscriberInfo 320
GetSubscriberInfoByOption 333
GetSubscriberList 336
GetTombStoneSubscriberList 343
GetVersion 94

H

home DNs 58
hot desking 398
hot desking (BG) 399
hunt groups
    overview 40
    phone services 621
    terminals 620

I

importing subscriber data 758
incoming call name delivery 391
InterceptList
    get 100
interface (WSDL) 61
interface release control 61
interface release control rules 61
interface support 33
interface version support 33
interface WSDL rules 61

J

jobs 45
    background 658
    cancelling 661
    retrieving a job list 660
    retrieving a job list by option 660
    retrieving information 658

K

keyset information
    call forward LED indication 57
    keyset use (enum) 55
    line appearance 56
    manual hold timer 56
keyset operation (BG) 399
keyset operation mode 57

L

location code 47

M

main numbers 52
MDR
    See message detail recording.
media gateways 45
media servers 488
message detail recording 53
Modify Class Of Restrictions 709
Modify DNM Conversion 599
Modify DNM Definition 571, 582
Modify DNM LocalToll 607
Modify DNM Normalization 591
Modify DNM Prefix 562
Modify Traffic Type 705, 736
ModifyDestCodeInfo 527
ModifyDestination 155, 174
modifying
    destination code information 527
    hunt group terminal information 640
    PNP extension 556
    PNP location codes 551
    prefix access codes 540
ModifyMlhgTermInfo 640
ModifyPnpExtension 556
ModifyPrefixAccessCode 540
music on hold 401

N

Network Management Interfaces via SOAP 117
No_SVC 58
number modification
    concepts 46
    SOAP/XML interface 560

O

office codes 57
One Number Service 402
OpenScape Voice Assistant
    RTP Configuration Parameteres 882
optimized parameter 51
order entry screens 32
originating numbering plan name (FromNpld) 50
outgoing caller ID suppression 391

P

paging 72
parameters
    call forward LED indication 57
    display number modification table 50
    display number prefix table 48
    skip digit (SK) 47
    unique level (UL) 47

parking lots 479
PNP Extension 554
PNPs
    See private number plans. 54
prefix access codes 535
prefix parameter 49
primitive data definitions 66
privacy type parameter 51
private numbering plans
    concepts 54
    location codes 550
    SOAP/XML interfaces 469
profile only subscribers 34
public/private network access code (PNAC) parameter 49

Q

Query Class Of Restrictions 710
Query Class Of Restrictions Info List 711
Query COR Usage List 712
Query DNM Conversion 602
Query DNM Definition 574, 575, 584
Query DNM LocalToll 610
Query DNM Normalization 594
Query DNM Prefix 565
Query DNM PrefixList 566
Query Traffic Type 707

R

RateArea
    get list 99
recent change logging
    in SOAP export 40
    in SOAP mass provisioning 40
    in the SOAP server 38
remote activation call forward 403
remote call forward 404
RemoveRateArea 159
removing old jobs 753
requirements
    CRM 32
    interface support 33
    system 31
    web server 32
result code information 878
result codes 881
result text settings 881
result text strings 880
ResultTOY 881
retrieving
    a list of jobs 660
    BG attendant numbers 440
    BG info by options 436
    BG list 437
    BG-related information 436
    destination code list 532
    destination service list 534
    hunt group terminal data 643
    individual prefix access codes 549
    job information 658
    jobs by option 660
    numbering plan
        information 521
        list 519
    PNP location codes 553
    prefix access code list 546
return code information 73

S

security 31
selective call forward 406
selective caller accept 405
selective caller reject 407
serial ringing 408
service activation code table 55
silent monitoring 418
silent monitoring protection 418
silent monitoring tag 418
simultaneous ringing 409
SIP endpoints 58
SIP network servers 59
SIP phones connection 53
sizing 31
skip digit parameter 47
SOAP messages 913
SOAP server 23
    process exit 750
    process startup 751
SOAP/XML conventions 65
Speed Call (Speed Dial) (BG) 412
standards support 33
station restrictions(BG) 410
station speed call (BG) 411
SubmgtSchedule process exit 751
subnet E911 LIN administration 53
subscrber profile data 24
subscriber connection info messages
    update subscriber connection info request 930
subscriber info sample messages
    get subscriber info request 919
    get subscriber info response 919
subscriber management mass provisioning
    importing subscriber data 758
subscriber provisioning 23, 878

Subscriber Provisioning Network Architecture Layout 24
subscriber service interactions 24
subscribers
  concepts 34
subscribers profile only 34
support
  interface version 33
  standards 33
supported subscribers 31
system requirements 31

T

terminals 620
terminating numbering plan name (ToNpld) 50
time zones 938
toll restrictions 414
traffic measurements
  business groups 54
Traffic Types and Classes of Restrictions 703, 731
traffic volumes 31
TranslateDialedDigits 110
type of number (TON) parameter 49, 50

U

unique level parameter 47
update subscriber account management info sample messages
  update subscriber account management info request 926
  update subscriber account management info response 927
update subscriber account user info sample messages
  update subscriber account user info request 928
  update subscriber account user info response 928
update subscriber DN sample messages
  update subscriber DN request 932
  update subscriber DN response 932
update subscriber features sample messages
  update subscriber features request 923
  update subscriber features response 924
update subscriber PIC sample messages
  update subscriber PIC request 925
  update subscriber PIC response 926
update subscriber status sample messages
  update subscriber status request 929
  update subscriber status response 929
UpdateBGStatus 430
UpdateBgSubnet 458
UpdateConnectionInfo 365
UpdateEpCapabilities 651
UpdateEpFeatures 653
UpdateEpMgtInfo 648
UpdateEpPics 651
UpdateFeatureProfile 614
UpdateKeysetInfo 356
UpdateSubscriberAccountMgtInfo 347
UpdateSubscriberAccountUserInfo 352
UpdateSubscriberBGInfo 350
UpdateSubscriberCapabilities 354
UpdateSubscriberDn 359
UpdateSubscriberFeatureProfile 366
UpdateSubscriberFeatures 361
UpdateSubscriberPICs 347
UpdateSubscriberStatus 346
Utilities
  GetCallingLocationList 101
  GetInterceptList 100
  GetRateAreaList 99
  TranslateDialedDigits 110
utilities
  Get COSList 99
  GetCallingLocationList 100, 101
  GetConfigData 95
  GetFeatureList 94
  GetInterceptList 100
  GetVersion 94
  TranslateDialedDigits 110, 111

W

web services description language
  definition 913
  interfacing to the system 913
  sample messages 913
WSDL
  See web services description language.