



QualNet 6.1

Developer Model Library

September 2012



Scalable Network Technologies, Inc.

6100 Center Drive, Suite 1250
Los Angeles, CA 90045

Phone: 310-338-3318
Fax: 310-338-7213

<http://www.scalable-networks.com>

Copyright Information

© 2012 Scalable Network Technologies, Inc. All rights reserved.

QualNet and EXata are registered trademarks of Scalable Network Technologies, Inc.

All other trademarks and trade names used are property of their respective companies.

Scalable Network Technologies, Inc.
6100 Center Drive, Suite 1250
Los Angeles, CA 90045
Phone: 310-338-3318
Fax: 310-338-7213

<http://www.scalable-networks.com>

Table of Contents

| | | |
|---|---------------------------------|----|
| Chapter 1 | Overview of Model Library | 1 |
| 1.1 List of Models in the Library | | 1 |
| 1.2 Conventions Used | | 3 |
| 1.2.1 Format for Command Line Configuration | | 3 |
| 1.2.1.1 General Format of Parameter Declaration | | 3 |
| 1.2.1.2 Precedence Rules | | 4 |
| 1.2.1.3 Parameter Description Format | | 5 |
| 1.2.2 Format for GUI Configuration | | 8 |
| Chapter 2 | MAC Layer Models..... | 13 |
| 2.1 802.3 LAN/Ethernet | | 14 |
| 2.1.1 Description | | 14 |
| 2.1.2 Features and Assumptions | | 14 |
| 2.1.2.1 Implemented Features | | 14 |
| 2.1.2.2 Omitted Features..... | | 14 |
| 2.1.2.3 Assumptions and Limitations..... | | 14 |
| 2.1.3 Command Line Configuration | | 15 |
| 2.1.4 GUI Configuration | | 17 |
| 2.1.5 Statistics | | 19 |
| 2.1.5.1 File Statistics | | 19 |
| 2.1.5.2 Database Statistics..... | | 21 |
| 2.1.5.3 Dynamic Statistics | | 21 |
| 2.1.6 Scenarios Included in QualNet..... | | 21 |
| 2.1.7 References..... | | 22 |
| 2.2 Abstract Link MAC Model | | 23 |
| 2.2.1 Description | | 23 |
| 2.2.2 Command Line Configuration | | 23 |
| 2.2.3 GUI Configuration | | 25 |

| | |
|--|-----------|
| 2.2.3.1 Configuring Abstract Link MAC for Symmetric Links | 25 |
| 2.2.3.2 Configuring Abstract Link MAC for Asymmetric Links | 27 |
| 2.2.3.3 Configuring Statistics Parameters | 29 |
| 2.2.4 Statistics | 29 |
| 2.2.4.1 File Statistics | 29 |
| 2.2.4.2 Database Statistics | 30 |
| 2.2.4.3 Dynamic Statistics | 31 |
| 2.3 Abstract Satellite Model | 32 |
| 2.3.1 Description | 32 |
| 2.3.2 Command Line Configuration | 32 |
| 2.3.3 GUI Configuration | 33 |
| 2.3.4 Statistics | 35 |
| 2.3.5 Sample Scenario | 35 |
| 2.3.5.1 Scenario Description | 35 |
| 2.3.5.2 Command Line Configuration | 36 |
| 2.3.5.3 GUI Configuration | 36 |
| 2.3.6 Scenarios Included in QualNet..... | 37 |
| 2.4 Address Resolution Protocol (ARP)..... | 38 |
| 2.4.1 Description | 38 |
| 2.4.2 Features and Assumptions | 38 |
| 2.4.2.1 Implemented Features | 38 |
| 2.4.2.2 Omitted Features..... | 38 |
| 2.4.2.3 Assumptions and Limitations..... | 38 |
| 2.4.3 Supplemental Information | 39 |
| 2.4.4 Command Line Configuration | 39 |
| 2.4.4.1 Format of the MAC Address Configuration File..... | 40 |
| 2.4.4.2 Format of the ARP Static Cache File | 41 |
| 2.4.5 GUI Configuration | 43 |
| 2.4.6 Statistics | 47 |
| 2.4.7 Scenarios Included in QualNet..... | 47 |
| 2.4.8 References..... | 48 |
| 2.5 Logical Link Control (LLC) Protocol | 49 |
| 2.5.1 Description | 49 |
| 2.5.2 Features and Assumptions | 49 |
| 2.5.2.1 Implemented Features | 49 |
| 2.5.2.2 Omitted Features..... | 49 |
| 2.5.2.3 Assumptions and Limitations..... | 49 |
| 2.5.3 Command Line Configuration | 49 |
| 2.5.4 GUI Configuration | 50 |
| 2.5.5 References..... | 51 |

| | | |
|--|-----------------------------------|-----------|
| Chapter 3 | Network Layer Models | 52 |
| 3.1 Fixed Communications Model | 53 | |
| 3.1.1 Description | 53 | |
| 3.1.2 Features and Assumptions | 53 | |
| 3.1.2.1 Implemented Features | 53 | |
| 3.1.2.2 Omitted Features..... | 53 | |
| 3.1.2.3 Assumptions and Limitations..... | 53 | |
| 3.1.3 Command Line Configuration | 53 | |
| 3.1.4 GUI Configuration | 54 | |
| 3.2 Internet Control Message Protocol (ICMP)..... | 57 | |
| 3.2.1 Description | 57 | |
| 3.2.2 Features and Assumptions | 57 | |
| 3.2.2.1 Implemented Features | 57 | |
| 3.2.2.2 Omitted Features..... | 58 | |
| 3.2.2.3 Assumptions and Limitations..... | 59 | |
| 3.2.3 Command Line Configuration | 59 | |
| 3.2.4 GUI Configuration | 63 | |
| 3.2.5 Statistics..... | 69 | |
| 3.2.6 Scenarios Included in QualNet..... | 70 | |
| 3.2.7 References..... | 70 | |
| 3.3 Internet Control Message Protocol version 6 (ICMPv6)..... | 72 | |
| 3.3.1 Description | 72 | |
| 3.3.2 Command Line Configurations..... | 72 | |
| 3.3.3 GUI Configuration | 73 | |
| 3.3.4 Statistics..... | 73 | |
| 3.3.5 References..... | 74 | |
| 3.4 Internet Group Management Protocol (IGMP)..... | 75 | |
| 3.4.1 Description | 75 | |
| 3.4.2 Features and Assumptions | 76 | |
| 3.4.2.1 Implemented Features | 76 | |
| 3.4.2.2 Omitted Features..... | 76 | |
| 3.4.2.3 Assumptions and Limitations..... | 76 | |
| 3.4.3 Command Line Configuration | 76 | |
| 3.4.4 GUI Configuration | 78 | |
| 3.4.5 Statistics | 82 | |
| 3.4.5.1 File Statistics | 83 | |
| 3.4.5.2 Database Statistics..... | 83 | |
| 3.4.5.3 Dynamic Statistics | 83 | |
| 3.4.6 Scenarios Included in QualNet..... | 84 | |
| 3.4.7 References..... | 84 | |
| 3.5 Internet Protocol - Dual IP | 85 | |

| | |
|--|------------|
| 3.5.1 Description | 85 |
| 3.5.1.1 Dual IP Layer Operation..... | 85 |
| 3.5.1.2 Configured Tunneling Mechanism..... | 85 |
| 3.5.1.3 6to4 Automatic Tunneling Mechanism | 86 |
| 3.5.2 Features and Assumptions | 86 |
| 3.5.2.1 Implemented Features | 86 |
| 3.5.2.2 Omitted Features..... | 86 |
| 3.5.2.3 Assumptions and Limitations..... | 86 |
| 3.5.3 Command Line Configuration | 87 |
| 3.5.3.1 Format of the Tunnel Configuration File..... | 88 |
| 3.5.4 GUI Configuration | 89 |
| 3.5.5 Statistics | 92 |
| 3.5.6 Scenarios Included in QualNet..... | 92 |
| 3.5.7 References..... | 93 |
| 3.6 Internet Protocol version 4 (IPv4)..... | 94 |
| 3.6.1 Description | 94 |
| 3.6.2 Command Line Configuration | 94 |
| 3.6.3 GUI Configuration | 95 |
| 3.6.4 Statistics | 97 |
| 3.6.4.1 File Statistics | 97 |
| 3.6.4.2 Database Statistics..... | 99 |
| 3.6.4.3 Dynamic Statistics | 99 |
| 3.6.5 References..... | 99 |
| 3.7 Internet Protocol version 6 (IPv6)..... | 100 |
| 3.7.1 Description | 100 |
| 3.7.2 Features and Assumptions | 101 |
| 3.7.2.1 Implemented Features | 101 |
| 3.7.2.2 Omitted Features..... | 101 |
| 3.7.2.3 Assumptions and Limitations..... | 101 |
| 3.7.3 Command Line Configuration | 101 |
| 3.7.4 GUI Configuration | 102 |
| 3.7.5 Statistics | 104 |
| 3.7.6 Scenarios Included in QualNet..... | 105 |
| 3.7.7 References..... | 105 |
| 3.8 Neighbor Discovery Protocol..... | 106 |
| 3.8.1 Description | 106 |
| 3.8.2 Features and Assumptions | 107 |
| 3.8.2.1 Implemented Features | 107 |
| 3.8.2.2 Omitted Features..... | 107 |
| 3.8.2.3 Assumptions and Limitations..... | 107 |
| 3.8.3 Command Line Configuration | 108 |
| 3.8.4 GUI Configuration | 108 |

| | |
|---|------------|
| 3.8.5 Statistics..... | 108 |
| 3.8.6 Scenarios Included in QualNet..... | 109 |
| 3.8.7 References..... | 109 |
| Chapter 4 Unicast Routing Protocol Models | 110 |
| 4.1 Bellman-Ford Routing Protocol..... | 111 |
| 4.1.1 Description | 111 |
| 4.1.2 Command Line Configuration | 111 |
| 4.1.3 GUI Configuration | 112 |
| 4.1.4 Statistics..... | 113 |
| 4.2 Routing Information Protocol next generation (RIPng)..... | 114 |
| 4.2.1 Description | 114 |
| 4.2.2 Command Line Configuration | 114 |
| 4.2.3 GUI Configuration | 115 |
| 4.2.4 Statistics..... | 117 |
| 4.2.4.1 File Statistics | 117 |
| 4.2.4.2 Dynamic Statistics..... | 117 |
| 4.2.5 Scenarios Included in QualNet..... | 118 |
| 4.3 Routing Information Protocol/Routing Information Protocol version 2 (RIP/ RIPv2) | 119 |
| 4.3.1 Description | 119 |
| 4.3.2 Command Line Configuration | 119 |
| 4.3.3 GUI Configuration | 120 |
| 4.3.4 Statistics..... | 122 |
| 4.3.4.1 File Statistics | 122 |
| 4.3.4.2 Dynamic Statistics..... | 122 |
| 4.3.5 Scenarios Included in QualNet..... | 123 |
| 4.4 Static and Default Routes..... | 124 |
| 4.4.1 Description | 124 |
| 4.4.2 Command Line Configuration | 124 |
| 4.4.2.1 Static Routes | 124 |
| 4.4.2.2 Default Routes..... | 124 |
| 4.4.2.3 Format of Static and Default Route Files | 125 |
| 4.4.3 GUI Configuration | 126 |
| 4.4.4 Scenarios Included in QualNet..... | 128 |
| Chapter 5 Multicast Routing Protocol Models | 129 |
| 5.1 Static Multicast Routes..... | 130 |
| 5.1.1 Description | 130 |
| 5.1.2 Command Line Configuration | 130 |

| | |
|---|------------|
| 5.1.2.1 Format of the Static Multicast Route File | 130 |
| 5.1.3 GUI Configuration | 131 |
| 5.1.4 Statistics..... | 132 |
| 5.1.5 Scenarios Included in QualNet..... | 133 |
| Chapter 6 Queues and Scheduler Models..... | 134 |
| 6.1 Class-Based Queueing (CBQ)..... | 135 |
| 6.1.1 Description | 135 |
| 6.1.2 Features and Assumptions | 135 |
| 6.1.2.1 Implemented Features | 135 |
| 6.1.2.2 Omitted Features..... | 135 |
| 6.1.2.3 Assumptions and Limitations..... | 135 |
| 6.1.3 Command Line Configuration | 135 |
| 6.1.3.1 Format of the Link Sharing Structure File..... | 137 |
| 6.1.4 GUI Configuration | 138 |
| 6.1.5 Statistics..... | 140 |
| 6.1.6 References..... | 140 |
| 6.2 First-In First-Out (FIFO) Queue | 141 |
| 6.2.1 Description | 141 |
| 6.2.2 Command Line Configuration | 141 |
| 6.2.3 GUI Configuration | 142 |
| 6.2.4 Statistics..... | 144 |
| 6.2.5 Scenarios Included in QualNet..... | 145 |
| 6.3 Random Early Detection (RED) Queue | 146 |
| 6.3.1 Description | 146 |
| 6.3.2 Features and Assumptions | 146 |
| 6.3.2.1 Implemented Features | 146 |
| 6.3.2.2 Omitted Features..... | 146 |
| 6.3.2.3 Assumptions and Limitations..... | 146 |
| 6.3.3 Command Line Configuration | 146 |
| 6.3.4 GUI Configuration | 148 |
| 6.3.5 Statistics..... | 150 |
| 6.3.6 Scenarios Included in QualNet..... | 151 |
| 6.3.7 References..... | 151 |
| 6.4 Random Early Detection with In/Out (RIO) Queue | 152 |
| 6.4.1 Description | 152 |
| 6.4.2 Command Line Configuration | 152 |
| 6.4.3 GUI Configuration | 156 |
| 6.4.4 Statistics..... | 159 |
| 6.4.5 Scenarios Included in QualNet..... | 160 |
| 6.4.6 References..... | 160 |

| | |
|--|------------|
| 6.5 Round Robin Scheduler | 161 |
| 6.5.1 Description | 161 |
| 6.5.2 Command Line Configuration | 161 |
| 6.5.3 GUI Configuration | 161 |
| 6.5.4 Statistics..... | 163 |
| 6.5.5 Scenarios Included in QualNet..... | 163 |
| 6.6 Self-Clocked Fair Queueing (SCFQ) Scheduler | 164 |
| 6.6.1 Description | 164 |
| 6.6.2 Features and Assumptions | 164 |
| 6.6.2.1 Implemented Features | 164 |
| 6.6.2.2 Omitted Features..... | 164 |
| 6.6.2.3 Assumptions and Limitations..... | 164 |
| 6.6.3 Command Line Configuration | 164 |
| 6.6.4 GUI Configuration | 165 |
| 6.6.5 Statistics..... | 167 |
| 6.6.6 Scenarios Included in QualNet..... | 168 |
| 6.6.7 References..... | 168 |
| 6.7 Strict Priority Scheduler | 169 |
| 6.7.1 Description | 169 |
| 6.7.2 Command Line Configuration | 169 |
| 6.7.3 GUI Configuration | 169 |
| 6.7.4 Statistics..... | 171 |
| 6.7.5 Scenarios Included in QualNet..... | 171 |
| 6.8 Weighted Fair Queueing (WFQ) Scheduler..... | 172 |
| 6.8.1 Description | 172 |
| 6.8.2 Features and Assumptions | 172 |
| 6.8.2.1 Implemented Features | 172 |
| 6.8.2.2 Omitted Features..... | 172 |
| 6.8.2.3 Assumptions and Limitations..... | 172 |
| 6.8.3 Command Line Configuration | 172 |
| 6.8.4 GUI Configuration | 173 |
| 6.8.5 Statistics..... | 175 |
| 6.8.6 Scenarios Included in QualNet..... | 176 |
| 6.8.7 References..... | 176 |
| 6.9 Weighted RED (WRED) Queue..... | 177 |
| 6.9.1 Description | 177 |
| 6.9.2 Command Line Configuration | 177 |
| 6.9.3 GUI Configuration | 180 |
| 6.9.4 Statistics..... | 183 |
| 6.9.5 Scenarios Included in QualNet..... | 184 |
| 6.9.6 References..... | 184 |
| 6.10 Weighted Round Robin (WRR) Scheduler | 186 |

| | |
|---|------------|
| 6.10.1 Description | 186 |
| 6.10.2 Features and Assumptions | 186 |
| 6.10.2.1 Implemented Features | 186 |
| 6.10.2.2 Omitted Features..... | 186 |
| 6.10.2.3 Assumptions and Limitations..... | 186 |
| 6.10.3 Command Line Configuration | 186 |
| 6.10.4 GUI Configuration | 187 |
| 6.10.5 Statistics..... | 189 |
| 6.10.6 Scenarios Included in QualNet..... | 189 |
| 6.10.7 References..... | 190 |
| Chapter 7 Transport Layer Models | 191 |
| 7.1 Abstract Transmission Control Protocol (Abstract TCP)..... | 192 |
| 7.1.1 Description | 192 |
| 7.1.2 Features and Assumptions | 192 |
| 7.1.2.1 Implemented Features | 192 |
| 7.1.2.2 Omitted Features..... | 192 |
| 7.1.2.3 Assumptions and Limitations..... | 192 |
| 7.1.3 Command Line Configuration | 192 |
| 7.1.4 GUI Configuration | 195 |
| 7.1.5 Statistics..... | 199 |
| 7.1.6 Scenarios Included in QualNet..... | 199 |
| 7.2 Multicast Dissemination Protocol | 200 |
| 7.2.1 Description | 200 |
| 7.2.2 Features and Assumptions | 201 |
| 7.2.2.1 Implemented Features | 201 |
| 7.2.2.2 Omitted Features..... | 201 |
| 7.2.2.3 Assumptions and Limitations..... | 201 |
| 7.2.3 Supplemental Information | 202 |
| 7.2.4 Command Line Configuration | 202 |
| 7.2.4.1 MDP Parameters for the Scenario Configuration File | 202 |
| 7.2.4.2 Running Forward-App with MDP | 204 |
| 7.2.4.3 Format of the MDP Profile FIlle..... | 204 |
| 7.2.5 GUI Configuration | 209 |
| 7.2.6 Statistics..... | 213 |
| 7.2.7 References..... | 214 |
| 7.3 Transmission Control Protocol (TCP)..... | 215 |
| 7.3.1 Description | 215 |
| 7.3.2 Features and Assumptions | 215 |
| 7.3.2.1 Implemented Features | 215 |
| 7.3.2.2 Omitted Features..... | 216 |

| | |
|---|------------|
| 7.3.2.3 Assumptions and Limitations..... | 216 |
| 7.3.3 Command Line Configuration | 216 |
| 7.3.3.1 Format of the TCP Dump (ASCII) File..... | 219 |
| 7.3.4 GUI Configuration | 221 |
| 7.3.5 Statistics..... | 229 |
| 7.3.5.1 File Statistics | 229 |
| 7.3.5.2 Database Statistics..... | 230 |
| 7.3.5.3 Dynamic Statistics..... | 231 |
| 7.3.6 Scenarios Included in QualNet..... | 231 |
| 7.3.7 References..... | 231 |
| 7.4 User Datagram Protocol (UDP)..... | 233 |
| 7.4.1 Description | 233 |
| 7.4.2 Command Line Configuration | 233 |
| 7.4.3 GUI Configuration | 233 |
| 7.4.4 Statistics | 234 |
| 7.4.4.1 File Statistics | 234 |
| 7.4.4.2 Database Statistics..... | 236 |
| 7.4.4.3 Dynamic Statistics..... | 236 |
| 7.4.5 References..... | 236 |
| Chapter 8 Application Layer Models | 237 |
| 8.1 Background Traffic Model..... | 238 |
| 8.1.1 Description | 238 |
| 8.1.2 Command Line Configuration | 238 |
| 8.1.2.1 Format of the Background Traffic File..... | 238 |
| 8.1.3 GUI Configuration | 241 |
| 8.1.4 Statistics..... | 247 |
| 8.1 Constant Bit Rate (CBR) Traffic Generator..... | 248 |
| 8.1.1 Description | 248 |
| 8.1.2 Features and Assumptions | 248 |
| 8.1.2.1 Implemented Features | 248 |
| 8.1.2.2 Omitted Features..... | 248 |
| 8.1.2.3 Assumptions and Limitations..... | 248 |
| 8.1.3 Command Line Configuration | 248 |
| 8.1.4 GUI Configuration | 252 |
| 8.1.5 Statistics..... | 257 |
| 8.1.5.1 File Statistics | 257 |
| 8.1.5.2 Database Statistics..... | 258 |
| 8.1.5.3 Dynamic Statistics..... | 258 |
| 8.2 File Transfer Protocol (FTP)..... | 259 |
| 8.2.1 Description | 259 |

| | |
|--|------------|
| 8.2.2 Command Line Configuration | 259 |
| 8.2.3 GUI Configuration | 260 |
| 8.2.4 Statistics..... | 262 |
| 8.2.4.1 File Statistics | 262 |
| 8.2.4.2 Database Statistics..... | 263 |
| 8.2.4.3 Dynamic Statistics | 263 |
| 8.2.5 References..... | 264 |
| 8.3 File Transfer Protocol/Generic (FTP/Generic)..... | 265 |
| 8.3.1 Description | 265 |
| 8.3.2 Command Line Configuration | 265 |
| 8.3.3 GUI Configuration | 268 |
| 8.3.4 Statistics..... | 270 |
| 8.3.4.1 File Statistics | 270 |
| 8.3.4.2 Database Statistics..... | 271 |
| 8.3.4.3 Dynamic Statistics | 271 |
| 8.4 HyperText Transfer Protocol (HTTP)..... | 272 |
| 8.4.1 Description | 272 |
| 8.4.2 Features and Assumptions | 272 |
| 8.4.2.1 Implemented Features | 272 |
| 8.4.2.2 Omitted Features..... | 272 |
| 8.4.2.3 Assumptions and Limitations..... | 272 |
| 8.4.3 Command Line Configuration | 272 |
| 8.4.4 GUI Configuration | 274 |
| 8.4.5 Statistics..... | 276 |
| 8.4.5.1 File Statistics | 276 |
| 8.4.5.2 Database Statistics..... | 277 |
| 8.4.5.3 Dynamic Statistics | 277 |
| 8.4.6 Sample Scenario | 278 |
| 8.4.6.1 Scenario Description | 278 |
| 8.4.6.2 Command Line Configuration..... | 278 |
| 8.4.7 References..... | 278 |
| 8.5 Lookup Traffic Generator..... | 279 |
| 8.5.1 Description | 279 |
| 8.5.2 Command Line Configuration | 279 |
| 8.5.3 GUI Configuration | 281 |
| 8.5.4 Statistics..... | 284 |
| 8.6 Multicast Constant Bit Rate (MCBR) Traffic Generator..... | 285 |
| 8.6.1 Description | 285 |
| 8.6.2 Command Line Configuration | 285 |
| 8.6.3 GUI Configuration | 288 |
| 8.6.4 Statistics..... | 293 |
| 8.6.4.1 File Statistics | 293 |

| | |
|---|------------|
| 8.6.4.2 Database Statistics..... | 294 |
| 8.6.4.3 Dynamic Statistics..... | 294 |
| 8.7 Super Application Traffic Generator | 295 |
| 8.7.1 Description | 295 |
| 8.7.2 Limitations and Assumptions | 295 |
| 8.7.3 Command Line Configuration | 295 |
| 8.7.3.1 Super Application Parameters Specified in Application Configuration File | 296 |
| 8.7.3.2 Super Application Parameters Specified in the Scenario Configuration File.. | 304 |
| 8.7.4 GUI Configuration | 306 |
| 8.7.4.1 Configuring a Client-Server Super Application Session | 306 |
| 8.7.4.2 Configuring a Single Host Super Application Session..... | 327 |
| 8.7.4.3 Configuring Statistics Parameters | 328 |
| 8.7.5 Statistics | 328 |
| 8.7.5.1 File Statistics | 328 |
| 8.7.5.2 Database Statistics..... | 330 |
| 8.7.5.3 Dynamic Statistics | 330 |
| 8.8 Telecommunications Network (TELNET)..... | 332 |
| 8.8.1 Description | 332 |
| 8.8.2 Command Line Configuration | 332 |
| 8.8.3 GUI Configuration | 333 |
| 8.8.4 Statistics..... | 335 |
| 8.9 Traffic Generator (Traffic-Gen) | 336 |
| 8.9.1 Description | 336 |
| 8.9.2 Command Line Configuration | 336 |
| 8.9.3 GUI Configuration | 344 |
| 8.9.3.1 Configuring a Client-Server Traffic-Gen Session | 344 |
| 8.9.3.2 Configuring a Single Host Traffic-Gen Session..... | 364 |
| 8.9.3.3 Configuring Statistics Parameters | 366 |
| 8.9.4 Statistics | 366 |
| 8.9.4.1 File Statistics | 366 |
| 8.9.4.2 Database Statistics..... | 367 |
| 8.9.4.3 Dynamic Statistics | 367 |
| 8.9.5 Scenarios Included in QualNet..... | 368 |
| 8.10 Trace File-based Traffic Generator (Traffic-Trace) | 369 |
| 8.10.1 Description | 369 |
| 8.10.2 Command Line Configuration | 369 |
| 8.10.2.1 Format of the Traffic Trace File | 377 |
| 8.10.3 GUI Configuration | 377 |
| 8.10.3.1 Configuring a Client-Server Traffic-Trace Session..... | 377 |
| 8.10.3.2 Configuring a Single Host Traffic-Trace Session | 386 |

| | |
|---|------------|
| 8.10.3.3 Configuring Statistics Parameters | 387 |
| 8.10.4 Statistics | 387 |
| 8.11 Variable Bit Rate (VBR) Traffic Generator | 389 |
| 8.11.1 Description | 389 |
| 8.11.2 Command Line Configuration | 389 |
| 8.11.3 GUI Configuration | 392 |
| 8.11.4 Statistics | 396 |
| 8.11.4.1 File Statistics | 396 |
| 8.11.4.2 Database Statistics..... | 397 |
| 8.11.4.3 Dynamic Statistics..... | 397 |
| Chapter 9 Multi-layer Models | 398 |
| 9.1 Asynchronous Transfer Mode (ATM) | 399 |
| 9.1.1 Description | 399 |
| 9.1.1.1 ATM Backbone..... | 399 |
| 9.1.1.2 IP over ATM | 400 |
| 9.1.2 Features and Assumptions | 401 |
| 9.1.2.1 Implemented Features | 401 |
| 9.1.2.2 Omitted Features..... | 402 |
| 9.1.2.3 Assumptions and Limitations..... | 402 |
| 9.1.3 Command Line Configuration | 402 |
| 9.1.3.1 Configuring the ATM Backbone | 402 |
| 9.1.3.2 Configuring IP over ATM | 406 |
| 9.1.4 GUI Configuration | 407 |
| 9.1.4.1 Configuring ATM Networks | 407 |
| 9.1.4.2 Configuring ATM Link Parameters | 410 |
| 9.1.4.3 Configuring Adaptation Layer Parameters | 412 |
| 9.1.4.4 Configuring ATM Layer 2 Parameters..... | 414 |
| 9.1.4.5 Configuring ATM Static Routes..... | 416 |
| 9.1.5 Statistics | 418 |
| 9.1.6 Sample Scenarios | 419 |
| 9.1.6.1 Standalone ATM Backbone Scenario | 419 |
| 9.1.6.2 IP over ATM Scenario | 423 |
| 9.1.7 Scenarios Included in QualNet..... | 426 |
| 9.1.8 References | 427 |
| Chapter 10 Interfaces | 428 |
| 10.1 AGI Satellite Toolkit (STK) Interface..... | 429 |
| 10.1.1 Description | 429 |
| 10.1.2 Features and Assumptions | 429 |

| | |
|--|------------|
| 10.1.2.1 Implemented Features | 429 |
| 10.1.2.2 Omitted Features..... | 429 |
| 10.1.2.3 Assumptions and Limitations..... | 429 |
| 10.1.3 Supplemental Information | 429 |
| 10.1.4 Integrating QualNet and STK | 430 |
| 10.1.5 Command Line Configuration | 430 |
| 10.1.6 GUI Configuration | 431 |
| 10.1.7 Statistics | 432 |
| Chapter 11 Miscellaneous Models..... | 433 |
| 11.1 Faults | 434 |
| 11.1.1 Description | 434 |
| 11.1.2 Command Line Configuration | 434 |
| 11.1.2.1 Format of the Fault Configuration..... | 434 |
| 11.1.3 GUI Configuration | 438 |
| 11.1.4 Runtime Visualization..... | 446 |
| 11.1.5 Statistics..... | 446 |
| 11.1.6 Scenarios Included in QualNet..... | 446 |
| 11.2 File-based Node Placement Model..... | 447 |
| 11.2.1 Description | 447 |
| 11.2.2 Command Line Configuration | 447 |
| 11.2.2.1 Format of the Node Position File..... | 447 |
| 11.2.3 GUI Configuration | 448 |
| 11.2.3.1 Using Node Placement Wizard | 449 |
| 11.2.3.2 Configuring Individual Node Placement Parameters..... | 450 |
| 11.3 Grid Node Placement Model | 452 |
| 11.3.1 Description | 452 |
| 11.3.2 Command Line Configuration | 452 |
| 11.3.3 GUI Configuration | 452 |
| 11.3.3.1 Using Node Placement Wizard | 453 |
| 11.3.3.2 Configuring Individual Node Placement Parameters..... | 454 |
| 11.4 Random Node Placement Model | 456 |
| 11.4.1 Description | 456 |
| 11.4.2 Command Line Configuration | 456 |
| 11.4.3 GUI Configuration | 456 |
| 11.4.3.1 Using Node Placement Wizard | 456 |
| 11.4.3.2 Configuring Individual Node Placement Parameters..... | 458 |
| 11.5 Uniform Node Placement Model..... | 460 |
| 11.5.1 Description | 460 |
| 11.5.2 Command Line Configuration | 460 |
| 11.5.3 GUI Configuration | 460 |

| | |
|--|-----|
| 11.5.3.1 Using Node Placement Wizard | 461 |
| 11.5.3.2 Configuring Individual Node Placement Parameters..... | 462 |

1

Overview of Model Library

1.1 List of Models in the Library

The models described in the Developer Model Library are listed in [Table 1-1](#).

TABLE 1-1. Developer Library Models

| Model Name | Model Type | Section Number |
|---|-----------------------|------------------------------|
| 802.3 LAN/Ethernet | MAC Layer | Section 2.1 |
| Abstract Link MAC | MAC Layer | Section 2.2 |
| Abstract Satellite Model | MAC Layer | Section 2.3 |
| Abstract Transmission Control Protocol (Abstract TCP) | Transport Layer | Section 7.1 |
| Address Resolution Protocol (ARP) | MAC Layer | Section 2.4 |
| AGI Satellite Toolkit (STK) | Interface | Section 10.1 |
| Asynchronous Transfer Mode (ATM) | Multi-layer | Section 9.1 |
| Background Traffic Model | Application Layer | Section 8.1 |
| Bellman-Ford Routing Protocol | Routing Protocol | Section 4.1 |
| Class-Based Queuing (CBQ) | Queues and Schedulers | Section 6.1 |
| Constant Bit Rate (CBR) Traffic Generator | Application Layer | Section 8.1 |
| Faults | Miscellaneous | Section 11.1 |
| File Transfer Protocol (FTP) | Application Layer | Section 8.2 |
| File Transfer Protocol/Generic (FTP/Generic) | Application Layer | Section 8.3 |
| File-based Node Placement Model | Miscellaneous | Section 11.2 |
| First-In First-Out (FIFO) Queue | Queues and Schedulers | Section 6.2 |
| Fixed Communications Model | Network Layer | Section 3.1 |
| Grid Node Placement Model | Miscellaneous | Section 11.3 |
| Hypertext Transfer Protocol (HTTP) | Application Layer | Section 8.4 |
| Internet Control Message Protocol (ICMP) | Network Layer | Section 3.2 |
| Internet Control Message Protocol version 6 (ICMPv6) | Network Layer | Section 3.3 |
| Internet Group Management Protocol (IGMP) | Network Layer | Section 3.4 |
| Internet Protocol - Dual IP | Network Layer | Section 3.5 |

TABLE 1-1. Developer Library Models (Continued)

| Model Name | Model Type | Section Number |
|---|-----------------------|------------------------------|
| Internet Protocol version 4 (IPv4) | Network Layer | Section 3.6 |
| Internet Protocol version 6 (IPv6) | Network Layer | Section 3.7 |
| Logical Link Control (LLC) Protocol | MAC Layer | Section 2.5 |
| Lookup Traffic Generator | Application Layer | Section 8.5 |
| Multicast Constant Bit Rate (MCBR) Traffic Generator | Application Layer | Section 8.6 |
| Multicast Dissemination Protocol (MDP) | Transport Layer | Section 7.2 |
| Neighbor Discovery Protocol | Network Layer | Section 3.8 |
| Random Early Detection (RED) Queue | Queues and Schedulers | Section 6.3 |
| Random Early Detection with In/Out (RIO) Queue | Queues and Schedulers | Section 6.4 |
| Random Node Placement Model | Miscellaneous | Section 11.4 |
| Round Robin Scheduler | Queues and Schedulers | Section 6.5 |
| Routing Information Protocol next generation (RIPng) | Routing Protocol | Section 4.2 |
| Routing Information Protocol/Routing Information Protocol version 2 (RIP/RIPv2) | Routing Protocol | Section 4.3 |
| Self-Clocked Fair Queueing (SCFQ) Scheduler | Queues and Schedulers | Section 6.6 |
| Static and Default Routes | Routing Protocol | Section 4.4 |
| Static Multicast Routes | Routing Protocol | Section 5.1 |
| Strict Priority Scheduler | Queues and Schedulers | Section 6.7 |
| Super Application Traffic Generator | Application Layer | Section 8.7 |
| Telecommunications Network (TELNET) | Application Layer | Section 8.8 |
| Trace File-based Traffic Generator (Traffic-Trace) | Application Layer | Section 8.10 |
| Traffic Generator (Traffic-Gen) | Application Layer | Section 8.9 |
| Transmission Control Protocol (TCP) | Transport Layer | Section 7.3 |
| Uniform Node Placement Model | Miscellaneous | Section 11.5 |
| User Datagram Protocol (UDP) | Transport Layer | Section 7.4 |
| Variable Bit Rate (VBR) Traffic Generator | Application Layer | Section 8.11 |
| Weighted Fair Queuing (WFQ) Scheduler | Queues and Schedulers | Section 6.8 |
| Weighted RED (WRED) Queue | Queues and Schedulers | Section 6.9 |
| Weighted Round Robin (WRR) Scheduler | Queues and Schedulers | Section 6.10 |

1.2 Conventions Used

1.2.1 Format for Command Line Configuration

This section describes the general format for specifying parameters in input files, the precedence rules for parameters, and the conventions used in the description of command line configuration for each model.

1.2.1.1 General Format of Parameter Declaration

The general format for specifying a parameter in an input file is:

```
[<Qualifier>] <Parameter Name> [<Index>] <Parameter Value>
```

where

<Qualifier>

The qualifier is optional and defines the scope of the parameter declaration. The scope can be one of the following: Global, Node, Subnet, and Interface. Multiple instances of a parameter with different qualifiers can be included in an input file. Precedence rules (see [Section 1.2.1.2](#)) determine the parameter value for a node or interface.

Global:

The parameter declaration is applicable to the entire scenario (to all nodes and interfaces), subject to precedence rules. The scope of a parameter declaration is global if the qualifier is not included in the declaration.

Example:

```
MAC-PROTOCOL MACDOT11
```

Node:

The parameter declaration is applicable to specified nodes, subject to precedence rules. The qualifier for a node-level declaration is a list of space-separated node IDs or a range of node IDs (specified by using the keyword `thru`) enclosed in square brackets.

Example:

```
[5 thru 10] MAC-PROTOCOL MACDOT11
```

Subnet:

The parameter declaration is applicable to all interfaces in specified subnets, subject to precedence rules. The qualifier for a subnet-level declaration is a space-separated list of subnet addresses enclosed in square brackets. A subnet address can be specified in the IP dot notation or in the QualNet N syntax.

Example:

```
[N8-1.0 N2-1.0] MAC-PROTOCOL MACDOT11
```

Interface:

The parameter declaration is applicable to specified interfaces. The qualifier for an interface-level declaration is a space-separated list of subnet addresses enclosed in square brackets.

Example:

```
[192.168.2.1 192.168.2.4] MAC-PROTOCOL MACDOT11
```

| | |
|-------------------|--|
| <Parameter Name> | Name of the parameter. |
| <Index> | Instance of the parameter to which this parameter declaration is applicable, enclosed in square brackets. This should be in the range 0 to $n-1$, where n is the number of instances of the parameter. |
| | The instance specification is optional in a parameter declaration. If an instance is not included, then the parameter declaration is applicable to all instances of the parameter, unless otherwise specified. |
| <Parameter Value> | Value of the parameter. |

Note: There should not be any spaces between the parameter name and the index.

Examples of parameter declarations in input files are:

| | |
|--|-----------------|
| PHY-MODEL | PHY802.11b |
| [1] PHY-MODEL | PHY802.11a |
| [N8-1.0] PHY-RX-MODEL | BER-BASED |
| [8 thru 10] ROUTING-PROTOCOL | RIP |
| [192.168.2.1 192.168.2.4] MAC-PROTOCOL | GENERICMAC |
| NODE-POSITION-FILE | ./default.nodes |
| PROPAGATION-CHANNEL-FREQUENCY [0] | 2.4e9 |
| [1 2] QUEUE-WEIGHT [1] | 0.3 |

Note In the rest of this document, we will not use the qualifier or the index in a parameter's description. Users should use a qualifier and/or index to restrict the scope of a parameter, as appropriate.

1.2.1.2 Precedence Rules

Parameters without Instances

If the parameter declarations do not include instances, then the following rules of precedence apply when determining the parameter values for specific nodes and interfaces:

Interface > Subnet > Node > Global

This can be interpreted as follows:

- The value specified for an interface takes precedence over the value specified for a subnet, if any.
- The value specified for a subnet takes precedence over the value specified for a node, if any.
- The value specified for a node takes precedence over the value specified for the scenario (global value), if any.

Parameters with Instances

If the parameter declarations are a combination of declarations with and without instances, then the following precedence rules apply (unless otherwise stated):

Interface[i] > Subnet[i] > Node[i] > Global[i] > Interface > Subnet > Node > Global

This can be interpreted as follows:

- Values specified for a specific instance (at the interface, subnet, node, or global level) take precedence over values specified without the instance.

- For values specified for the same instance at different levels, the following precedence rules apply:
 - The value specified for an interface takes precedence over the value specified for a subnet, if any, if both declarations are for the same instance.
 - The value specified for a subnet takes precedence over the value specified for a node, if any, if both declarations are for the same instance.
 - The value specified for a node takes precedence over the value specified for the scenario (global value), if any, if both declarations are for the same instance.

1.2.1.3 Parameter Description Format

In the Model Library, most parameters are described using a tabular format described below. The parameter description tables have three columns labeled “Parameter”, “Values”, and “Description”. [Table 1-2](#) shows the format of parameter tables. [Table 1-4](#) shows examples of parameter descriptions in this format.

TABLE 1-2. Parameter Table Format

| Parameter | Values | Description |
|------------------|-------------------|---------------|
| <Parameter Name> | <Type> | <Description> |
| <Designation> | [<Range>] | |
| <Scope> | [<Default Value>] | |
| [<Instances>] | [<Unit>] | |

Parameter Column

The first column contains the following entries:

- <Parameter Name>**: The first entry is the parameter name (this is the exact name of the parameter to be used in the input files).
- <Designation>**: This entry can be *Optional* or *Required*. These terms are explained below.
 - Optional**: This indicates that the parameter is optional and may be omitted from the configuration file. (If applicable, the default value for this parameter is included in the second column.)
 - Required**: This indicates that the parameter is mandatory and must be included in the configuration file.
- <Scope>**: This entry specifies the possible scope of the parameter, i.e., if the parameter can be specified at the global, node, subnet, or interface levels. Any combination of these levels is possible. If the parameter can be specified at all four levels, the keyword “All” is used to indicate that.

Examples of scope specification are:

Scope: All

Scope: Subnet, Interface

Scope: Global, Node

- <Instances>**: If the parameter can have multiple instances, this entry indicates the type of index. If the parameter can not have multiple instances, then this entry is omitted.

Examples of instance specification are:

Instances: channel number

Instances: interface index

Instances: queue index

Values Column

The second column contains the following information:

- **<Type>**: The first entry is the parameter type and can be one of the following: Integer, Real, String, Time, Filename, IP Address, Coordinates, Node-list, or List. If the type is a List, then all possible values in the list are enumerated below the word “List”. (In some cases, the values are listed in a separate table and a reference to that table is included in place of the enumeration.)

Table 1-3 shows the values a parameter can take for each type.

TABLE 1-3. Parameter Types

| Type | Description |
|------------|--|
| Integer | Integer value Examples: 2, 10 |
| Real | Real value Examples: 15.0, -23.5, 2.0e9 |
| String | String value Examples: TEST, SWITCH1 |
| Time | Time value expressed in QualNet time syntax (refer to <i>QualNet User's Guide</i>) Examples: 1.5S, 200MS, 10US |
| Filename | Name of a file in QualNet filename syntax (refer to <i>QualNet User's Guide</i>) Examples: ..\..\data\terrain\los-angeles-w (For Windows and UNIX) C:\snt\qualnet\6.1\scenarios\WF\WF.nodes (For Windows) /root/snt/qualnet/6.1/scenarios/WF/WF.nodes (For UNIX) |
| Path | Path to a directory in QualNet path syntax (refer to <i>QualNet User's Guide</i>) Examples: ..\..\data\terrain (For Windows and UNIX) C:\snt\qualnet\6.1\scenarios\default (For Windows) /root/snt/qualnet/6.1/scenarios/default (For UNIX) |
| IP Address | IPv4 or IPv6 address Examples: 192.168.2.1, 2000:0:0:0::1 |

TABLE 1-3. Parameter Types (Continued)

| Type | Description |
|--------------|--|
| IPv4 Address | IPv4 address Examples: 192.168.2.1 |
| IPv6 Address | IPv6 address Examples: 2000:0:0:0::1 |
| Coordinates | Coordinates in Cartesian or Lat-Lon-Alt system. The altitude is optional. Examples: (100, 200, 2.5), (-25.3478, 25.28976) |
| Node-list | List of node IDs separated by commas and enclosed in {" and "}. Examples: {2, 5, 10}, {1, 3 thru 6} |
| List | One of the enumerated values. Example: See the parameter MOBILITY in Table 1-4 . |

Note: If the parameter type is List, then options for the parameter available in QualNet and the commonly used model libraries are enumerated. Additional options for the parameter may be available if some other model libraries or addons are installed. These additional options are not listed in this document but are described in the corresponding model library or addon documentation.

- **<Range>**: This is an optional entry and is used if the range of values that a parameter can take is restricted. The permissible range is listed after the label “Range.” The range can be specified by giving the minimum value, the maximum value, or both. If the range of values is not restricted, then this entry is omitted.

If both the minimum and maximum values are specified, then the following convention is used to indicate whether the minimum and maximum values are included in the range:

| | |
|------------|-----------------------------|
| (min, max) | min < parameter value < max |
| [min, max) | min ≤ parameter value < max |
| (min, max] | min < parameter value ≤ max |
| [min, max] | min ≤ parameter value ≤ max |

min (or max) can be a parameter name, in which case it denotes the value of that parameter.

Examples of range specification are:

Range: ≥ 0

Range: (0.0, 1.0]

Range: [1, MAX-COUNT]

Range: [1S, 200S]

Note: If an upper limit is not specified in the range, then the maximum value that the parameter can take is the largest value of the type (integer, real, time) that can be stored in the system.

- **<Default>**: This is an optional entry which specifies the default value of an optional or conditional-optional parameter. The default value is listed after the label “*Default*”.
- **<Unit>**: This is an optional entry which specifies the unit for the parameter, if applicable. The unit is listed after the label “*Unit*”. Examples of units are: meters, dBm, slots.

Description Column

The third column contains a description of the parameter. The significance of different parameter values is explained here, where applicable. In some cases, references to notes, other tables, sections in the User’s Guide, or to other model libraries may be included here.

Table 1-4 shows examples of parameter descriptions using the format described above.

TABLE 1-4. Example Parameter Table

| Parameter | Values | Description |
|--|---|---|
| MOBILITY Optional Scope: Global, Node | List: <ul style="list-style-type: none">• NONE• FILE• GROUP-MOBILITY• RANDOM-WAYPOINT Default: NONE | Mobility model used for the node. If MOBILITY is set to NONE, then the nodes remain fixed in one place for the duration of the simulation. See Table 7-11 for a description of mobility models. |
| BACKOFF-LIMIT Required Scope: Subnet, Interface | Integer Range: [4, 10] Unit: slots | Upper limit of backoff interval after collision. A backoff interval is randomly chosen between 1 and this number following a collision. |
| IP-QUEUE-PRIORITY-QUEUE-SIZE Required Scope: All Instances: queue index | Integer Range: [1, 65535] Unit: bytes | Size of the output priority queue. |
| MAC-DOT11-DIRECTIONAL-ANTENNA-MODE Optional Scope: All | List <ul style="list-style-type: none">• YES• NO Default: NO | Indicates whether the radio is to use a directional antenna for transmission and reception. |

1.2.2 Format for GUI Configuration

The GUI configuration section for a model outlines the steps to configure the model using the GUI. The following conventions are used in the GUI configuration sections:

Path to a Parameter Group

As a shorthand, the location of a parameter group in a properties editor is represented as a path consisting of the name of the properties editor, name of the tab within the properties editor, name of the parameter group within the tab (if applicable), name of the parameter sub-group (if applicable), and so on.

Example

The following statement:

Go to **Default Device Properties Editor > Interfaces > Interface # > MAC Layer**

is equivalent to the following sequence of steps:

1. Open the Default Device Properties Editor for the node.
2. Click the **Interfaces** tab.
3. Expand the applicable Interface group.
4. Click the **MAC Layer** parameter group.

The above path is shown in [Figure 1-1](#).

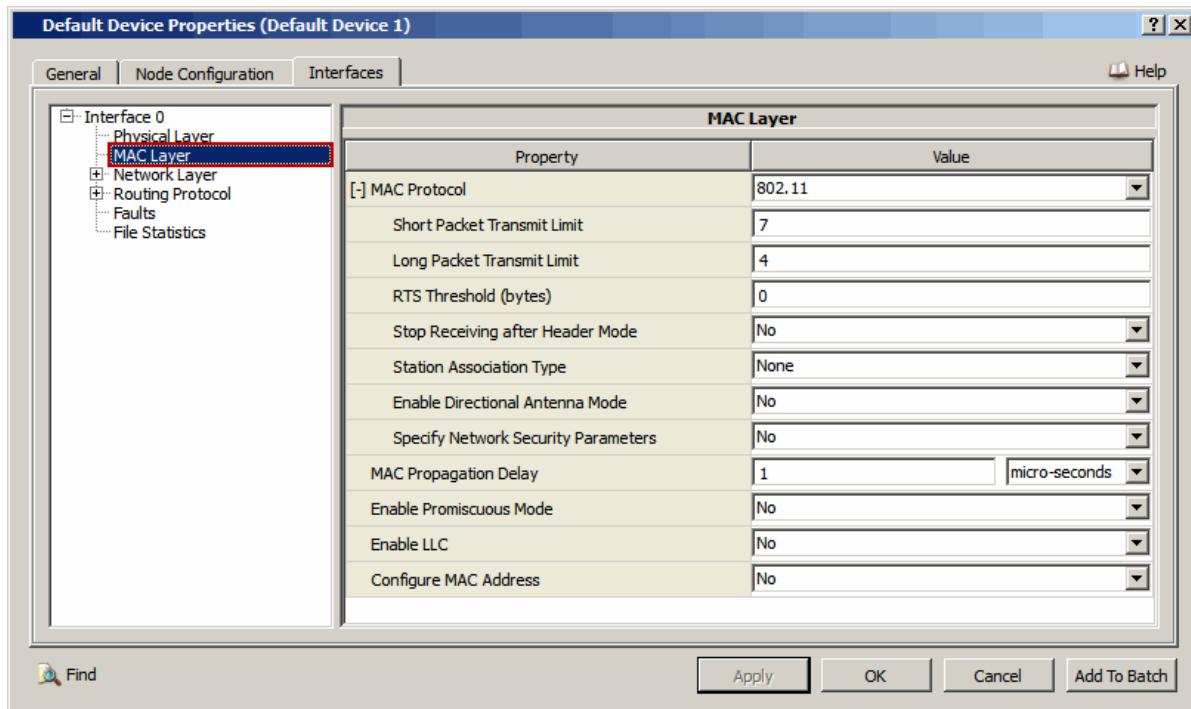


FIGURE 1-1. Path to a Parameter Group

Path to a Specific Parameter

As a shorthand, the location of a specific parameter within a parameter group is represented as a path consisting of all ancestor parameters and their corresponding values starting from the top-level parameter. The value of an ancestor parameter is enclosed in square brackets after the parameter name.

Example

The following statement:

Set MAC Protocol [= 802.11] > Station Association Type [= Dynamic] > Set Access Point [= Yes] > Enable Power Save Mode to Yes

is equivalent to the following sequence of steps:

1. Set **MAC Protocol** to *802.11*.
2. Set **Station Association Type** to *Dynamic*.
3. Set **Set Access Point** to *Yes*.
4. Set **Enable Power Save Mode** to *Yes*.

The above path is shown in [Figure 1-2](#).

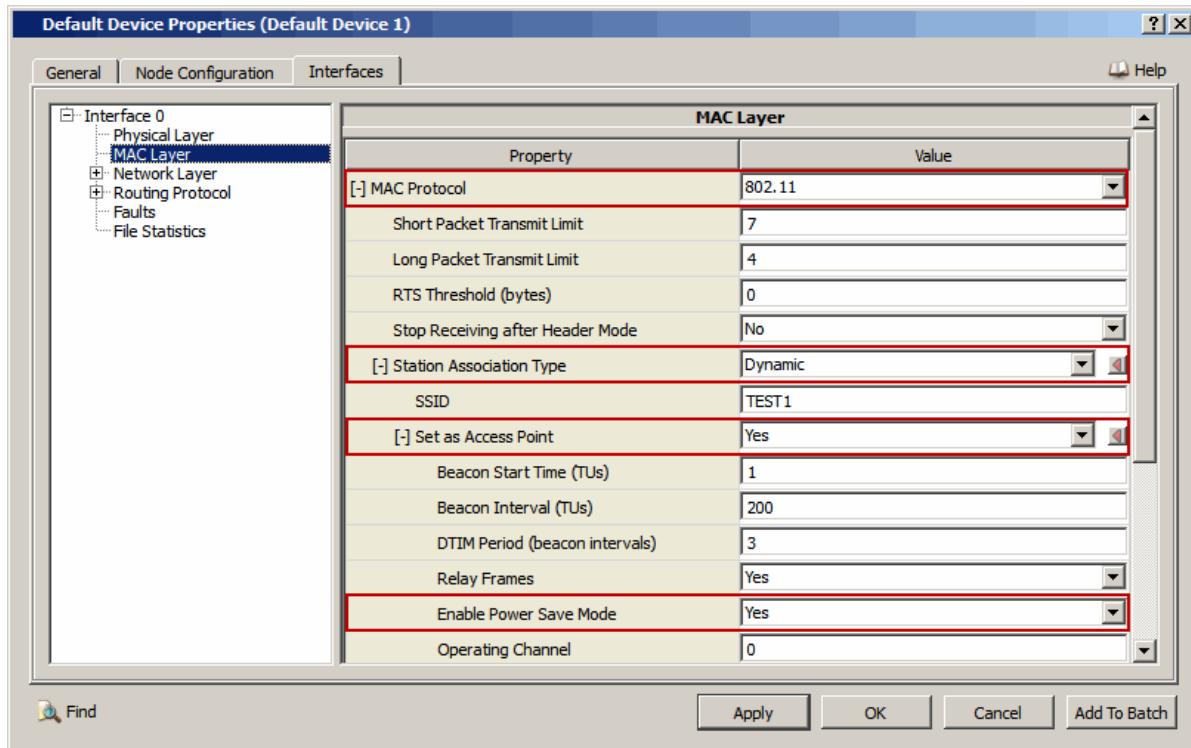


FIGURE 1-2. Path to a Specific Parameter

Parameter Table

GUI configuration of a model is described as a series of steps. Each step describes how to configure one or more parameters. Since the GUI display name of a parameter may be different from the name in the configuration file, each step also includes a table that shows the mapping between the GUI names and command line names of parameters configured in that step. For a description of a GUI parameter, see the description of the equivalent command line parameter in the command line configuration section.

The format of a parameter mapping table is shown in [Table 1-5](#).

TABLE 1-5. Mapping Table

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|--------------------|------------------------|-------------------------------|
| <GUI Display Name> | <Scope> | <Command Line Parameter Name> |

The first column, labeled “GUI Parameter”, lists the name of the parameter as it is displayed in the GUI.

The second column, labeled “Scope of GUI Parameter”, lists the level(s) at which the parameter can be configured. <Scope> can be any combination of: Global, Node, Subnet, Wired Subnet, Wireless Subnet, Point-to-point Link, and Interface.

[Table 1-6](#) lists the Properties Editors where parameters with different scopes can be set.

- Notes:**
1. Unless otherwise stated, the “Subnet” scope refers to “Wireless Subnet”.
 2. The scope column can also refer to Properties Editors for special devices and network components (such as ATM Device Properties Editor) which are not included in [Table 1-6](#).

TABLE 1-6. Properties Editors for Different Scopes

| Scope of GUI Parameter | Properties Editor |
|---------------------------|---|
| Global | Scenario Properties Editor |
| Node | Default Device Properties Editor (General and Node Configuration tabs) |
| Subnet Wireless Subnet | Wireless Subnet Properties Editor |
| Wired Subnet | Wired Subnet Properties Editor |
| Point-to-point Link | Point-to-point Link Properties Editor |
| Interface | Interface Properties Editor, Default Device Properties Editor (Interfaces tab) |

The third column, labeled “Command Line Parameter”, lists the equivalent command line parameter.

- Note:** For some parameters, the scope may be different in command line and GUI configurations (a parameter may be configurable at fewer levels in the GUI than in the command line).

[Table 1-7](#) is an example of a parameter mapping table.

TABLE 1-7. Example Mapping Table

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|--|------------------------|------------------------|
| Define Area | Node | OSPFv2-DEFINE-AREA |
| OSPFv2 Configuration File | Node | OSPFv2-CONFIG-FILE |
| Specify Autonomous System | Node | N/A |
| Configure as Autonomous System Boundary Router | Node | AS-BOUNDARY-ROUTER |
| Inject External Route | Node | N/A |
| Enable Stagger Start | Node | OSPFv2-STAGGER-START |

2 MAC Layer Models

This chapter describes features, configuration requirements and parameters, statistics, and scenarios for MAC Layer Models, and consists of the following sections:

- 802.3 LAN/Ethernet
- Abstract Link MAC
- Abstract Satellite Model
- Address Resolution Protocol (ARP)
- Logical Link Control (LLC) Protocol

2.1 802.3 LAN/Ethernet

The QualNet 802.3 LAN/Ethernet model is based on the IEEE 802.3, 2000 edition (incorporating IEEE Std 802.3, 1998 Edition, IEEE Std 802.3ac-1998, IEEE Std 802.3ab-1999, and IEEE Std 802.3ad-2000).

2.1.1 Description

The IEEE 802.3 LAN model models a single bus LAN of either 10 Mbps/1G (Gigabit Ethernet) or 100 Mbps/10G (Gigabit Fast Ethernet) hosts. QualNet supports both half-duplex and full-duplex modes of data transmission.

In half-duplex mode, data can flow between two stations in both directions, but in only one direction at a time. Half-duplex Ethernet uses the same basic CSMA/CD access mechanism as the 10 and 100 Mbps varieties of Ethernet, with the major exception of the slot time. The slot time in Gigabit Ethernet is modified to accommodate the special timing constraints which arise from the speed of the system.

In full-duplex mode, data can flow between two stations in both directions at the same time. Full-duplex Ethernet bypasses the normal CSMA/CD protocol to allow two stations to communicate over a point-to-point link. It effectively doubles the transfer rate by allowing each station to concurrently transmit and receive separate data streams. For example, a 10 Mbps full-duplex Ethernet station can transmit one 10 Mbps stream at the same time it receives a separate 10 Mbps stream. This provides an overall data transfer rate of 20 Mbps.

2.1.2 Features and Assumptions

This section describes the implemented features, omitted features, assumptions and limitations of the 802.3 LAN model.

2.1.2.1 Implemented Features

- Full-duplex Ethernet
- Full-duplex support for Switch
- Full-duplex support for VLAN
- Both full-duplex and half-duplex support for Gigabit Ethernet in addition to Ethernet and Fast Ethernet
- Full-duplex support for ARP

2.1.2.2 Omitted Features

- Pause frames
- Link aggregation

2.1.2.3 Assumptions and Limitations

None.

2.1.3 Command Line Configuration

To select 802.3 LAN/Ethernet as the MAC protocol, include one of the following parameters in the scenario configuration (.config) file:

[<Qualifier>] MAC-PROTOCOL MAC802.3

or

[<Qualifier>] LINK-MAC-PROTOCOL MAC802.3

The scope of this parameter declaration can be Global, Node, Subnet, or Interface. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

Note: To specify 802.3 LAN as the MAC protocol for a subnet defined by using the SUBNET statement, use the parameter MAC-PROTOCOL.

To specify 802.3 LAN as the MAC protocol for a subnet defined by using the LINK statement, use the parameter LINK-MAC-PROTOCOL.

802.3 LAN Parameters

802.3 LAN configuration parameters are described in [Table 2-1](#). See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 2-1. 802.3 LAN Parameters

| Parameter | Value | Description |
|---|--|-----------------------------------|
| SUBNET-DATA-RATE Optional (see note 1) Scope: All | Integer <i>Range:</i> ≥ 0 (see note 3) <i>Unit:</i> bits/second | Data rate for the subnet. |
| LINK-BANDWIDTH Optional (see note 2) Scope: All | Integer <i>Range:</i> ≥ 0 (see note 3) <i>Unit:</i> bits/second | Link bandwidth. |
| SUBNET-PROPAGATION-DELAY Optional (see note 1) Scope: All | Time <i>Range:</i> $\geq 0\text{S}$ (see note 4) | Propagation delay for the subnet. |
| LINK-PROPAGATION-DELAY Optional (see note 2) Scope: All | Time <i>Range:</i> $\geq 0\text{S}$ | Propagation delay for the link. |

TABLE 2-1. 802.3 LAN Parameters (Continued)

| Parameter | Value | Description |
|--|--|---|
| MAC802.3-MODE Optional Scope: All | List: <ul style="list-style-type: none">• HALF-DUPLEX• FULL-DUPLEX <i>Default:</i> HALF-DUPLEX | Duplex mode. If the subnet is defined using the LINK statement, then parameter MAC802.3-MODE can only be set to FULL-DUPLEX. If the subnet is defined using the SUBNET statement, then parameter MAC802.3-MODE can be set to FULL-DUPLEX only if there are two nodes in the subnet. |
| MAC-LAYER-STATISTICS Optional Scope: All | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Indicates whether statistics are collected for MAC protocols, including 802.3 LAN. |

- Notes:**
1. Parameters SUBNET-PROPAGATION-DELAY and SUBNET-DATA-RATE must be specified if the subnet is defined by using the SUBNET statement. These parameters must have the same value for all interfaces of a subnet.
 2. Parameters LINK-PROPAGATION-DELAY and LINK-BANDWIDTH must be specified if the LINK statement is used to connect two nodes. These parameters must have the same value for both interfaces of a link.
 3. For 802.3 LAN, SUBNET-DATA-RATE and LINK-BANDWIDTH can have only the following values (corresponding to 10M, 100M, 1G and 10G Ethernet): 10000000, 100000000, 1000000000, or 1000000000.
 4. The propagation delay should be smaller than half of the Ethernet slot time in order for nodes to detect collisions. The Ethernet slot time depends on data rate. The slot time of 10M Ethernet is 51200 nanoseconds and the propagation delay should be smaller than 25600 nanoseconds. The slot time of 100M Ethernet is 5120 nanoseconds and the propagation delay should be less than 2560 nanoseconds. The slot time of 1G and 10G Ethernet is 4096 nanoseconds and the propagation delay should be smaller than 2048 nanoseconds.

Examples of Parameter Usage

The following are examples of 802.3 LAN configuration:

1. The following lines show how to configure 802.3 LAN for a subnet in half-duplex mode:

```
SUBNET N2-192.0.2.0 {1, 2}
[N2-192.0.2.0] MAC-PROTOCOL           MAC802.3
[N2-192.0.2.0] SUBNET-DATA-RATE        1000000000
[N2-192.0.2.0] SUBNET-PROPAGATION-DELAY 2US
[N2-192.0.2.0] MAC-802.3-MODE          HALF-DUPLEX
```

2. The following lines show how to configure 802.3 LAN for a subnet in full-duplex mode:

```
SUBNET N2-192.0.2.0 {1, 2}
[N2-192.0.2.0] SUBNET-DATA-RATE      1000000000
[N2-192.0.2.0] SUBNET-PROPAGATION-DELAY 1US
[N2-192.0.2.0] MAC-PROTOCOL        MAC802.3
[N2-192.0.2.0] MAC-802.3-MODE       FULL-DUPLEX
```

3. The following lines show how to configure 802.3 LAN for a link in full-duplex mode:

```
LINK N2-192.0.2.0 {1, 2}
[N2-192.0.2.0] LINK-MAC-PROTOCOL      MAC802.3
[N2-192.0.2.0] LINK-BANDWIDTH        1000000000
[N2-192.0.2.0] LINK-PROPAGATION-DELAY 1US
[N2-192.0.2.0] MAC-802.3-MODE       FULL-DUPLEX
```

2.1.4 GUI Configuration

This section describes how to configure 802.3 LAN in the GUI.

Configuring 802.3 LAN Parameters

To configure the general 802.3 LAN parameters, perform the following steps:

1. Go to one of the following locations:

- To set properties at the subnet level, go to **Wired Subnet Properties Editor > MAC Layer**.
- To set properties for a specific point-to-point link, go to **Point-to-point Link Properties Editor > Point-to-point Link Properties > Routing Protocol**.
- To set properties at the interface level, go to one of the following locations:
 - **Interface Properties Editor > Interfaces > Interface # > MAC Layer**.
 - **Default Device Properties Editor > Interfaces > Interface # > MAC Layer**.

In this section, we show how to configure the general 802.3 LAN parameters in the Wired Subnet Properties Editor. Parameters can be set in the other properties editors in a similar way.

2. Set **MAC Protocol** to 802.3 and set the dependent parameters listed in [Table 2-2](#).

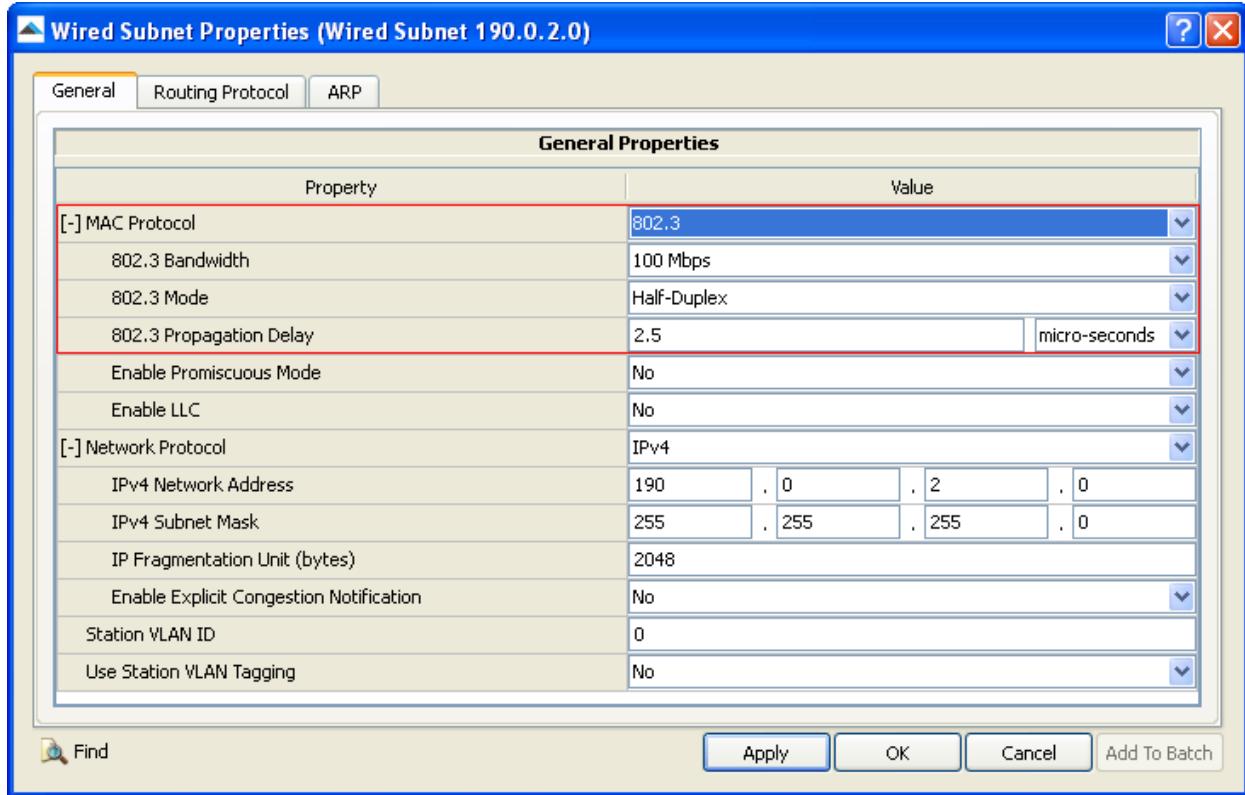


FIGURE 2-1. Setting 802.3 LAN Parameters

TABLE 2-2. Command Line Equivalent of 802.3 LAN Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-------------------------|--|--|
| 802.3 Bandwidth | Subnet, Point-to-point Link, Interface | SUBNET-DATA-RATE (for subnets) LINK-BANDWIDTH (for links) |
| 802.3 Mode | Subnet, Point-to-point Link, Interface | MAC802.3-MODE |
| 802.3 Propagation Delay | Subnet, Point-to-point Link, Interface | SUBNET-PROPAGATION-DELAY (for subnets) LINK-PROPAGATION-DELAY (for links) |

Configuring Statistics Parameters

Statistics for 802.3 LAN can be collected at the global, node, subnet and interface levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for 802.3 LAN, check the box labeled **MAC** in the appropriate properties editor.

TABLE 2-3. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|---------------------------------|------------------------|
| MAC | Global, Node, Subnet, Interface | MAC-LAYER-STATISTICS |

2.1.5 Statistics

This section describes the file, database, and dynamic statistics of the 802.3 LAN model.

2.1.5.1 File Statistics

Table 2-4 lists the 802.3 LAN statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 2-4. 802.3 LAN Statistics

| Statistic | Description |
|--|--|
| Unicast data frames sent to the phy layer (frames) | Total number of unicast data frames sent to the link. |
| Unicast data frames received from phy layer (frames) | Total number of unicast data frames received on the link. |
| Unicast control frames sent to the phy layer (frames) | Total number of unicast control frames sent to the link. |
| Unicast control frames received from phy layer (frames) | Total number of unicast control frames received on the link. |
| Unicast data bytes sent to the phy layer (bytes) | Total number of unicast data bytes sent to the link. |
| Unicast data bytes received from phy layer (bytes) | Total number of unicast data bytes received on the link. |
| Unicast control bytes sent to the phy layer (bytes) | Total number of unicast control bytes sent to the link. |
| Unicast control bytes received from phy layer (bytes) | Total number of unicast control bytes received on the link. |
| Average delay for Unicast packets in output queue at the mac layer (seconds) | Average delay in output queue at the MAC layer for unicast packets. |
| Average delay to gain access to medium at the mac layer for Unicast packets (seconds) | Average delay to gain access to medium at the MAC layer for unicast packets. |
| Average medium delay (transmission + propagation) at the mac layer for Unicast packets (seconds) | Average medium delay(transmission + propagation) at the MAC layer for unicast packets. |
| Average jitter at the mac layer for Unicast packets (seconds) | Average jitter at the MAC layer for the unicast packets. |
| Broadcast data frames sent to the phy layer (frames) | Total number of broadcast data frames sent to the link. |
| Broadcast data frames received from phy layer (frames) | Total number of broadcast data frames received on the link. |
| Broadcast control frames sent to the phy layer (frames) | Total number of broadcast control frames sent to the link. |
| Broadcast control frames received from phy layer (frames) | Total number of broadcast control frames received on the link. |
| Broadcast data bytes sent to the phy layer (bytes) | Total number of broadcast data bytes sent to the link. |

TABLE 2-4. 802.3 LAN Statistics (Continued)

| Statistic | Description |
|--|---|
| Broadcast data bytes received from phy layer (bytes) | Total number of broadcast data bytes received on the link. |
| Broadcast control bytes sent to the phy layer (bytes) | Total number of control bytes sent to the link. |
| Broadcast control bytes received from phy layer (bytes) | Total number of control bytes received on the link. |
| Average delay for Broadcast packets in output queue at the mac layer (seconds) | Average delay in output queue at the MAC layer for broadcast packets. |
| Average delay to gain access to medium at the mac layer for Broadcast packets (seconds) | Average delay to gain access to medium at the MAC layer for broadcast packets. |
| Average medium delay (transmission + propagation) at the mac layer for Broadcast packets (seconds) | Average medium delay (transmission + propagation) at the mac layer for broadcast packets |
| Average jitter at the mac layer for Broadcast packets (seconds) | Average jitter at the mac layer for broadcast packets. |
| Multicast data frames sent to the phy layer (frames) | Total number of multicast data frames sent to the link. |
| Multicast data frames received from phy layer (frames) | Total number of multicast data frames received on the link. |
| Multicast control frames sent to the phy layer (frames) | Total number of multicast control frames sent to the link. |
| Multicast control frames received from phy layer (frames) | Total number of multicast control frames received from the link. |
| Multicast data bytes sent to the phy layer (bytes) | Total number of multicast data bytes sent to the link. |
| Multicast data bytes received from phy layer (bytes) | Total number of multicast data bytes received on the link. |
| Multicast control bytes sent to the phy layer (bytes) | Total number of multicast control bytes sent to the link. |
| Multicast control bytes received from phy layer (bytes) | Total number of multicast control bytes received on the link. |
| Average delay for Multicast packets in output queue at the mac layer (seconds) | Average delay in output queue at the MAC layer for multicast packets. |
| Average delay to gain access to medium at the mac layer for Multicast packets (seconds) | Average delay to gain access to medium at the MAC layer for multicast packets. |
| Average medium delay (transmission + propagation) at the mac layer for Multicast packets (seconds) | Average medium delay (transmission + propagation) at the MAC layer for multicast packets. |
| Average jitter at the mac layer for Multicast packets (seconds) | Average jitter at the MAC layer for multicast packets. |
| Carried load at the mac layer (bits/second) | Carried load at the MAC layer. |
| Additional Statistics in Half-Duplex Mode | |
| Number of Retransmissions | Total number of retransmissions in half-duplex mode. |
| Number of Frames Dropped | Total number of frames dropped in half-duplex mode. |
| Additional Statistics in Full-Duplex Mode | |
| Number of Frames Dropped in Full-Duplex | Total number of frames dropped in full-duplex mode. |
| Number of Bytes Dropped in Full-Duplex | Total number of bytes dropped in full-duplex mode. |
| Channel Utilization by Full-Duplex | Channel utilization in full-duplex mode. See note. |

Note: Channel utilization in full-duplex mode is calculated as follows:

$$\text{channel-utilization} = \text{total-busy-time} / (\text{end-time} - \text{start-time})$$

where,

total-busy-time = total time for which the node transmits on the interface

end-time = End time. (This is the same as simulation time.)

start-time = Start time. (This is equal to 0)

2.1.5.2 Database Statistics

In addition to the file statistics, the 802.3 LAN model also enters statistics in various scenario statistics database tables. Refer to *QualNet Statistics Database User's Guide* for details.

2.1.5.3 Dynamic Statistics

No dynamic statistics are supported for the 802.3 LAN model.

2.1.6 Scenarios Included in QualNet

The QualNet distribution includes several sample scenarios for the 802.3 LAN model. All scenarios are located in the directory `QUALNET_HOME/scenarios/developer/802.3`. [Table 2-5](#) lists the sub-directory where each scenario is located.

TABLE 2-5. 802.3 LAN Scenarios Included in QualNet

| Scenario | Description |
|--------------------------------------|--|
| collision1-subnet1 | Shows the collision where node 2 starts to send its packet when the packet from node 1 has traveled half the way to node 2. |
| collision1-subnet2 | Shows the collision where node 2 sends its packet when first bit of the packet from node 1 is about to reach at node 2. |
| collision1-subnet3 | Shows the collision where node 2 starts to send its packets after the packets from node 1 have traveled half the way to node 2 in a subnet with high propagation delay. |
| collision2-subnet1 | Shows the occurrence of collision in a subnet when two different nodes have tried to transmit their frames simultaneously. |
| collision2-subnet2 | Shows the frame exchange between two subnets without any collision though probability of collision is present. |
| fullduplex/fulldup-double-throughput | Shows the performance of FULL-DUPLEX ETHERNET that doubles the throughput. |
| fullduplex/fulldup-mixed-duplexity | Shows the transmission and reception of frame in different subnets with one in Full Duplex transmission mode and the other in Half Duplex, the nodes being the same in both the subnets. |
| fullduplex/fulldup-switched-network1 | Shows the operation of FULL-DUPLEX ETHERNET in a totally switched network. |
| fullduplex/fulldup-switched-network2 | Shows the operation of FULL-DUPLEX ETHERNET in a mixed switched network. |
| fullduplex/fulldup-switched-vlan | Shows the performance of subnets in Full Duplex with switched vlan configured. |

TABLE 2-5. 802.3 LAN Scenarios Included in QualNet (Continued)

| Scenario | Description |
|-----------------|---|
| normal1-subnet1 | Shows the normal scenario where node 1 is transmitting and node 2 is receiving. |
| normal1-subnet2 | Shows the normal scenario where both node 1 and node 2 are transmitting to each other. But due to the difference in their frame origination time, collision does not arise in the subnet. |
| normal2-subnet | Shows a normal scenario where frames are transmitted in individual subnets without any collisions. |
| normal3-subnet1 | Shows two Ethernet subnets whether they can transmit packets through another type of subnet. |
| normal3-subnet2 | Shows the Ethernet subnet whether it can forward packets to different type of subnets. |

2.1.7 References

The QualNet MAC 802.3 model is based on the IEEE standards and the information available at the following URLs:

1. <http://standards.ieee.org/getieee802/802.3.html>
2. TechFest - Ethernet Technical Summary - Chapter 3,
<http://www.techfest.com/networking/lan/ethernet3.htm>
3. http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/ethernet.htm

2.2 Abstract Link MAC Model

2.2.1 Description

The Abstract Link MAC is an abstract MAC model for wired, wireless, and microwave point-to-point links.

2.2.2 Command Line Configuration

To select the Abstract Link MAC as the MAC protocol, include the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] LINK-MAC-PROTOCOL ABSTRACT
```

The scope of this parameter declaration can be Global, Node, Subnet, or Interface. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

Abstract Link MAC Parameters for Wired and Wireless Links

[Table 2-6](#) shows the configuration parameters Abstract Link MAC protocol for wired and wireless links. For Abstract Link MAC protocol for microwave links, refer to the Microwave Links section of *Wireless Model Library*. See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 2-6. Abstract Link MAC Parameters for Wired and Wireless Links

| Parameter | Value | Description |
|---|---|---|
| LINK-PROPAGATION-DELAY <i>Optional</i> <i>Scope:</i> All | Time <i>Range:</i> > 0s <i>Default:</i> 1ms | Time it takes a signal to travel from one node to the other node. Note: This parameter is required for a wired link, i.e., if LINK-PHY-TYPE is set to WIRED . It is not used for a wireless link, i.e., if LINK-PHY-TYPE is set to WIRELESS . |
| LINK-PROPAGATION-SPEED <i>Optional</i> <i>Scope:</i> Global <i>Instances:</i> channel index | Real <i>Range:</i> > 0.0 <i>Default:</i> 3.0e8 <i>Unit:</i> meters/sec | Signal propagation speed. Note: This parameter is used only for a wireless link. It is not used for a wired link. |
| LINK-BANDWIDTH <i>Required</i> <i>Scope:</i> All | Integer <i>Range:</i> ≥ 0 <i>Default:</i> 10 <i>Unit:</i> Mbps | Bandwidth of the link. The link is assumed to be full-duplex (data can be sent in both directions at the same time). |

TABLE 2-6. Abstract Link MAC Parameters for Wired and Wireless Links (Continued)

| Parameter | Value | Description |
|--|--|---|
| LINK-HEADER-SIZE-IN-BITS <i>Optional</i> Scope: All | Integer <i>Range:</i> > 0 <i>Default:</i> 224 <i>Unit:</i> bits | Header size. |
| LINK-GENERATE-AUTOMATIC-DEFAULT-ROUTE <i>Optional</i> Scope: All | List: • YES • NO <i>Default:</i> YES | Enables the automatic generation and population of a default route between the two end-points of the link in the IP forwarding table. If this parameter is set to YES, the two end-points of a link will be able to route packets to each other without any further configuration. If this parameter is set to NO, other measures must be taken to ensure that packets can be routed between the two connected nodes. |
| MAC-LAYER-STATISTICS <i>Optional</i> Scope: All | List: • YES • NO <i>Default:</i> NO | Specifies whether MAC layer statistics are enabled or disabled. |

Note: If any of the parameters LINK-BANDWIDTH, LINK-PROPAGATION-DELAY, LINK-HEADER-SIZE-IN-BITS, and LINK-GENERATE-AUTOMATIC-DEFAULT-ROUTE is not qualified, then it applies to both directions of the traffic flow. For asymmetric links, qualify the parameter with the interface address of the source node of the traffic.

Example of Parameter Usage

The following is an example of an asymmetric link configuration:

```

LINK N2-1.0 { 1, 2 }
LINK N2-2.0 { 2, 3 }
[1.1] LINK-BANDWIDTH      1544000
[1.2] LINK-BANDWIDTH      2374000
[1.1] LINK-PROPAGATION-DELAY 1MS
[1.2] LINK-PROPAGATION-DELAY 2MS
[2.1] LINK-BANDWIDTH      2156000
[2.2] LINK-BANDWIDTH      1234000
[2.1] LINK-PROPAGATION-DELAY 3MS
[2.2] LINK-PROPAGATION-DELAY 4MS
LINK-HEADER-SIZE-IN-BITS   200

```

The first LINK statement creates a point-to-point link between nodes 1 and 2. The second LINK statement creates a point-to-point link between nodes 2 and 3.

The link is full duplex, and asymmetric in propagation delay and bandwidth, i.e. the values of delay and bandwidth for a single link are different for different interfaces. This means that the values of bandwidth and delay of the link N2-1.0 from interface 1.1 to 1.2 are different from the bandwidth and delay values from interface 1.2 to 1.1.

2.2.3 GUI Configuration

This section describes how to configure Abstract Link MAC parameters for wired and wireless links in the GUI.

Note: To configure Abstract Link MAC parameters for a microwave link, refer to the Microwave Links section of *Wireless Model Library*.

2.2.3.1 Configuring Abstract Link MAC for Symmetric Links

To configure Abstract Link MAC parameters for a symmetric link, perform the following steps:

1. Go to the **Point-to-point Link Properties Editor > Point-to-point Link Properties > General**.
2. Set **MAC Protocol** to *Abstract Link MAC* and set the dependent parameters shown in Figure 2-2.

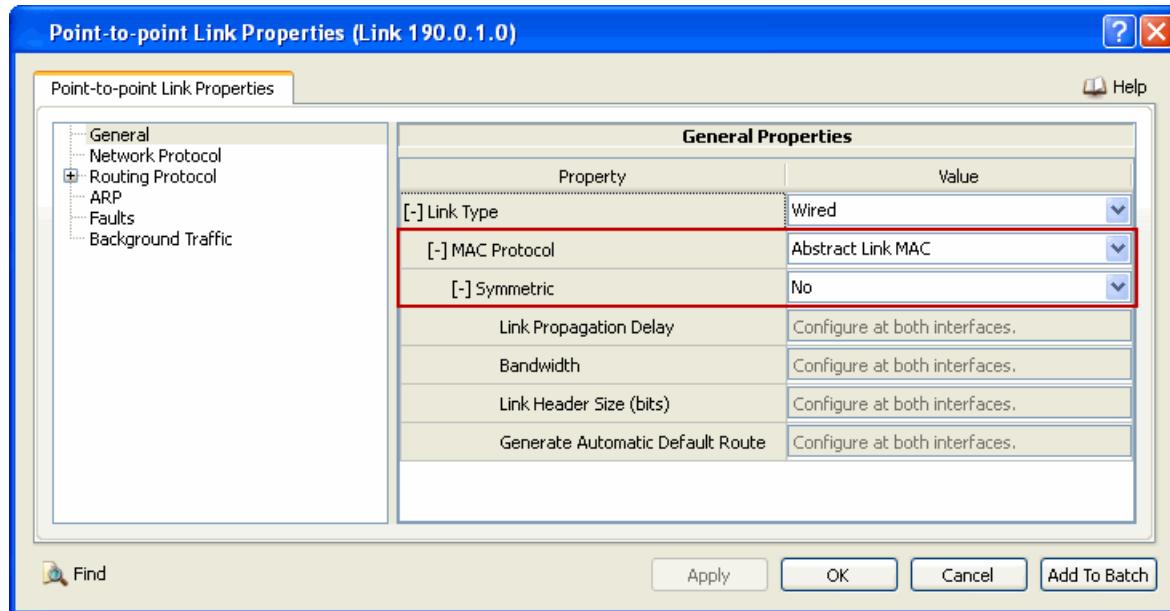


FIGURE 2-2. Setting Abstract Link MAC Parameters

Setting Parameters

- To configure the link as a symmetric link, set **Symmetric** to Yes; otherwise, set **Symmetric** to No.

3. If the link is a wired link (i.e., **Link Type** is set to *Wired*) and **Symmetric** is set to Yes, set the dependent parameters listed in [Table 2-7](#).

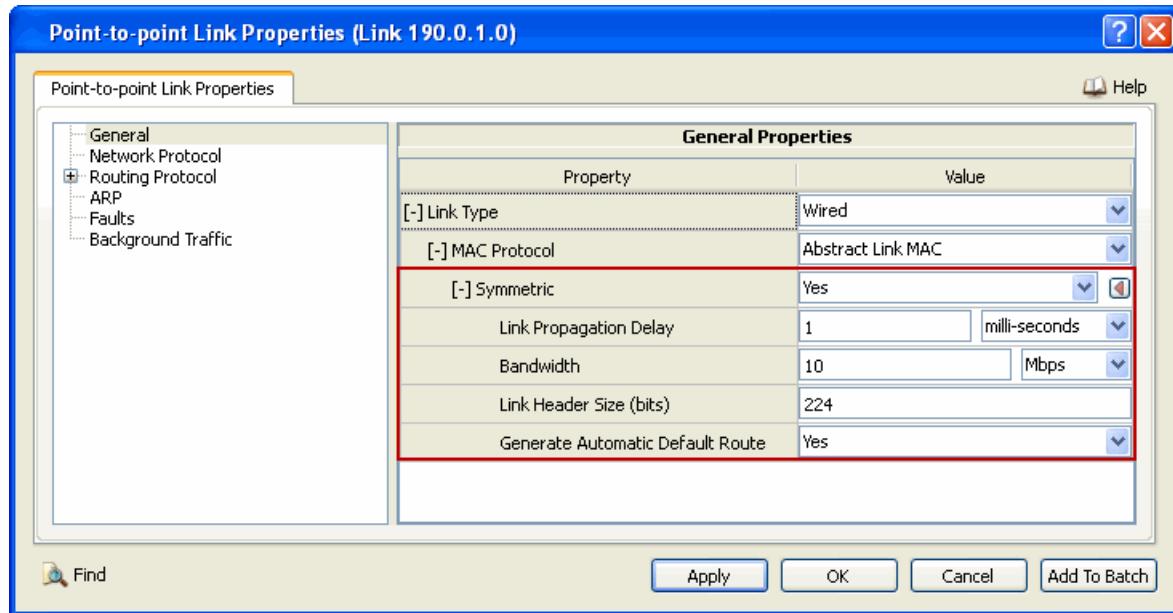


FIGURE 2-3. Setting Abstract Link MAC Parameters for a Symmetric Wired Link

TABLE 2-7. Command Line Equivalent of Symmetric Wired Link Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|----------------------------------|------------------------|---------------------------------------|
| Link Propagation Delay | Subnet, Interface | LINK-BANDWIDTH |
| Bandwidth | Subnet, Interface | LINK-PROPAGATION-DELAY |
| Link Header Size | Subnet, Interface | LINK-HEADER-SIZE-IN-BITS |
| Generate Automatic Default Route | Subnet, Interface | LINK-GENERATE-AUTOMATIC-DEFAULT-ROUTE |

4. If the link is a wireless link (i.e., **Link Type** is set to *Wireless*) and **Symmetric** is set to Yes, set the dependent parameters listed in [Table 2-8](#).

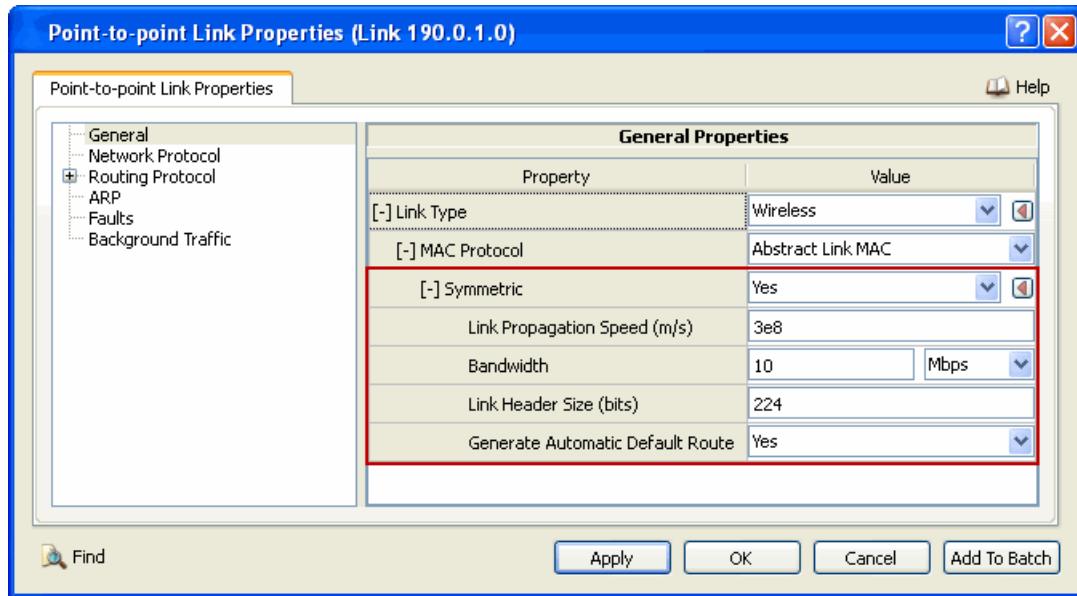


FIGURE 2-4. Setting Abstract Link MAC Parameters for a Symmetric Wireless Link

TABLE 2-8. Command Line Equivalent of Symmetric Wireless Link Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|----------------------------------|------------------------|---------------------------------------|
| Link Propagation Speed | Subnet, Interface | LINK-BANDWIDTH |
| Bandwidth | Subnet, Interface | LINK-PROPAGATION-DELAY |
| Link Header Size | Subnet, Interface | LINK-HEADER-SIZE-IN-BITS |
| Generate Automatic Default Route | Subnet, Interface | LINK-GENERATE-AUTOMATIC-DEFAULT-ROUTE |

2.2.3.2 Configuring Abstract Link MAC for Asymmetric Links

To configure Abstract Link MAC parameters for an asymmetric link, perform the following steps:

- Configure the link to be an asymmetric link, as described in [Section 2.2.3.1](#).
- For each interface of the link, configure the Abstract Link MAC parameters as follows:
 - Go to **Interface Properties Editor > Interfaces > Interface # > MAC Layer**.
 - If the link is a wired link, set the dependent parameters listed in [Table 2-7](#).

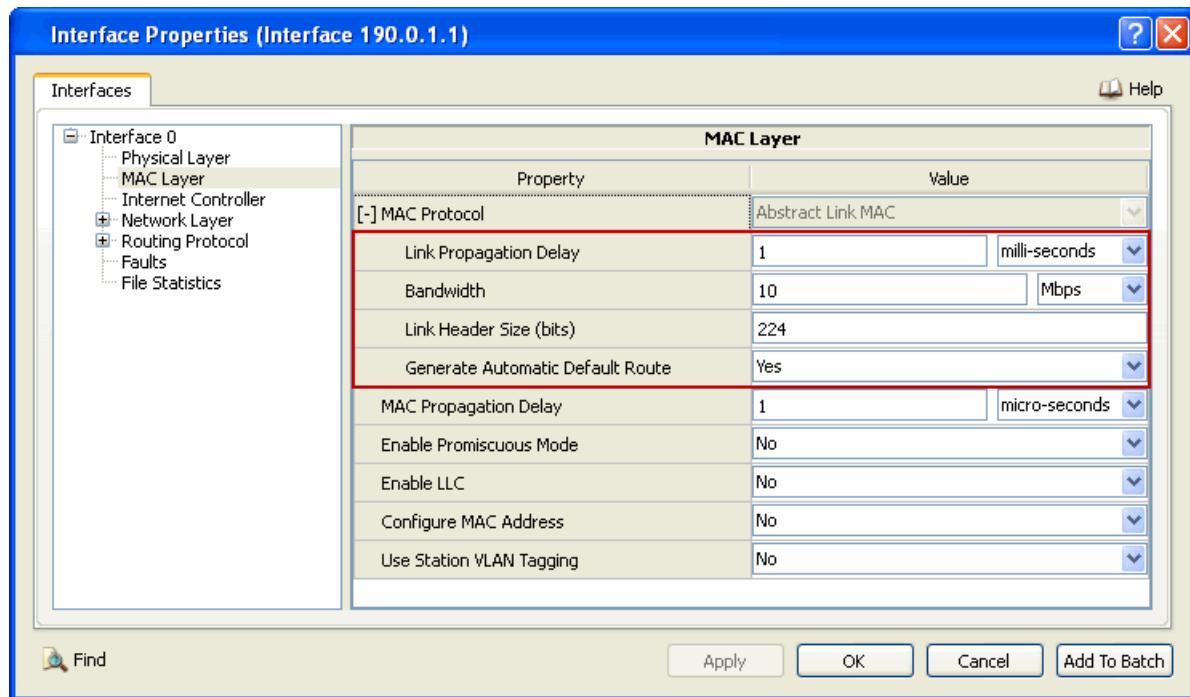


FIGURE 2-5. Setting Abstract Link MAC Parameters for an Asymmetric Wired Link

- c. If the link is a wireless link, set the dependent parameters listed in Table 2-8.

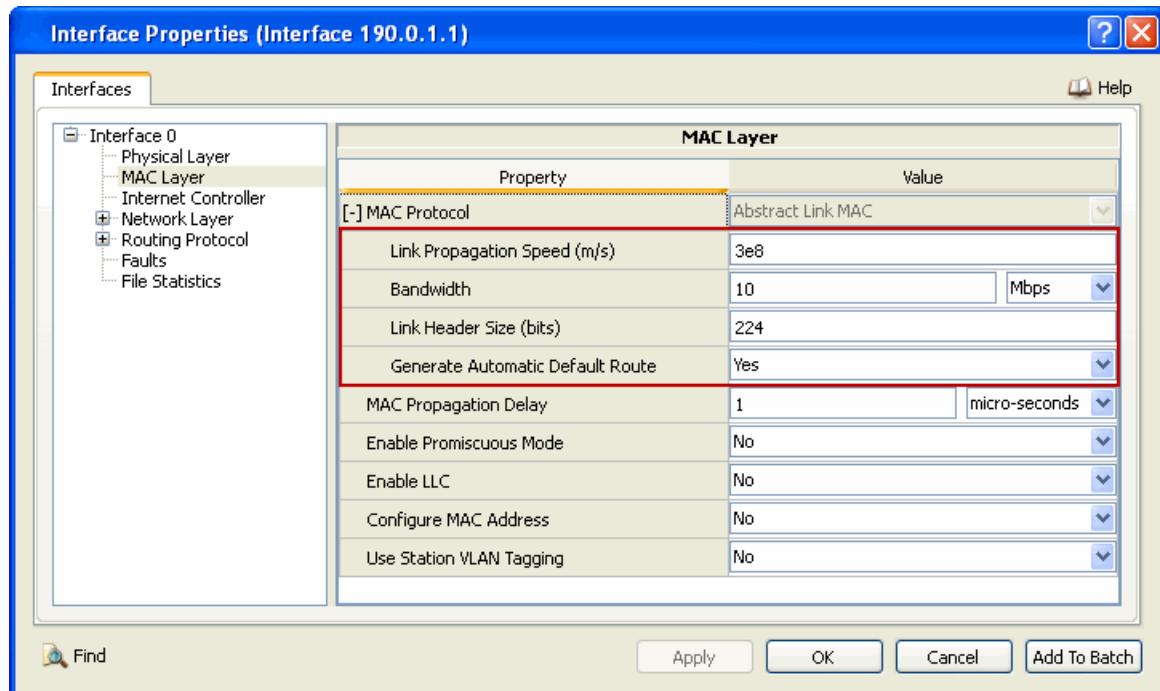


FIGURE 2-6. Setting Abstract Link MAC Parameters for an Asymmetric Wireless Link

2.2.3.3 Configuring Statistics Parameters

Statistics for the Abstract Link MAC model can be collected at the global, node, subnet, and interface levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for Abstract Link MAC, check the box labeled **MAC** in the appropriate properties editor.

TABLE 2-9. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|---------------------------------|------------------------|
| MAC | Global, Node, Subnet, Interface | MAC-LAYER-STATISTICS |

2.2.4 Statistics

This section describes the file, database, and dynamic statistics of the Abstract Link MAC model.

2.2.4.1 File Statistics

[Table 2-10](#) lists the Abstract Link MAC statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 2-10. Abstract Link MAC Statistics

| Statistics | Description |
|--|--|
| Destination | Shows the destination node ID for the link. |
| Link Utilization | Show the Link usage (see note) |
| Unicast data frames sent to the phy layer (frames) | Total number of unicast data frames sent to the link. |
| Unicast data frames received from phy layer (frames) | Total number of unicast data frames received on the link. |
| Unicast control frames sent to the phy layer (frames) | Total number of unicast control frames sent to the link. |
| Unicast control frames received from phy layer (frames) | Total number of unicast control frames received on the link. |
| Unicast data bytes sent to the phy layer (bytes) | Total number of unicast data bytes sent to the link. |
| Unicast data bytes received from phy layer (bytes) | Total number of unicast data bytes received on the link. |
| Unicast control bytes sent to the phy layer (bytes) | Total number of unicast control bytes sent to the link. |
| Unicast control bytes received from phy layer (bytes) | Total number of unicast control bytes received on the link. |
| Average delay for Unicast packets in output queue at the mac layer (seconds) | Average delay in output queue at the mac layer for unicast packets. |
| Average delay to gain access to medium at the mac layer for Unicast packets (seconds) | Average delay to gain access to medium at the MAC layer for unicast packets |
| Average medium delay (transmission + propagation) at the mac layer for Unicast packets (seconds) | Average medium delay(transmission + propagation) at the MAC layer for unicast packets. |
| Average jitter at the mac layer for Unicast packets (seconds) | Average jitter at the MAC layer for the unicast packets. |
| Broadcast data frames sent to the phy layer (frames) | Total number of broadcast data frames sent to the link. |
| Broadcast data frames received from phy layer (frames) | Total number of broadcast data frames received on the link. |

TABLE 2-10. Abstract Link MAC Statistics (Continued) (Continued)

| Statistics | Description |
|--|---|
| Broadcast control frames sent to the phy layer (frames) | Total number of broadcast control frames sent to the link. |
| Broadcast control frames received from phy layer (frames) | Total number of broadcast control frames received on the link. |
| Broadcast data bytes sent to the phy layer (bytes) | Total number of broadcast data bytes sent to the link. |
| Broadcast data bytes received from phy layer (bytes) | Total number of broadcast data bytes received on the link. |
| Broadcast control bytes sent to the phy layer (bytes) | Total number of control bytes sent to the link. |
| Broadcast control bytes received from phy layer (bytes) | Total number of control bytes received on the link. |
| Average delay for Broadcast packets in output queue at the mac layer (seconds) | Average delay in output queue at the MAC layer for broadcast packets. |
| Average delay to gain access to medium at the mac layer for Broadcast packets (seconds) | Average delay to gain access to medium at the MAC layer for broadcast packets. |
| Average medium delay (transmission + propagation) at the mac layer for Broadcast packets (seconds) | Average medium delay (transmission + propagation) at the MAC layer for broadcast packets. |
| Average jitter at the mac layer for Broadcast packets (seconds) | Average jitter at the MAC layer for broadcast packets. |
| Multicast data frames sent to the phy layer (frames) | Total number of multicast data frames sent to the link. |
| Multicast data frames received from phy layer (frames) | Total number of multicast data frames received on the link. |
| Multicast control frames sent to the phy layer (frames) | Total number of multicast control frames sent to the link. |
| Multicast control frames received from phy layer (frames) | Total number of multicast control frames received from the link. |
| Multicast data bytes sent to the phy layer (bytes) | Total number of multicast data bytes sent to the link. |
| Multicast data bytes received from phy layer (bytes) | Total number of multicast data bytes received on the link. |
| Multicast control bytes sent to the phy layer (bytes) | Total number of multicast control bytes sent to the link. |
| Multicast control bytes received from phy layer (bytes) | Total number of multicast control bytes received on the link. |
| Average delay for Multicast packets in output queue at the mac layer (seconds) | Average delay in output queue at the MAC layer for multicast packets. |
| Average delay to gain access to medium at the mac layer for Multicast packets (seconds) | Average delay to gain access to medium at the MAC layer for multicast packets. |
| Average medium delay (transmission + propagation) at the mac layer for Multicast packets (seconds) | Average medium delay (transmission + propagation) at the MAC layer for multicast packets. |
| Average jitter at the mac layer for Multicast packets (seconds) | Average jitter at the MAC layer for multicast packets. |
| Carried load at the mac layer (bits/second) | Carried load at the MAC layer. |

Note: If simulation time is 0, then the link utilization is also 0. Otherwise, it is given by the ratio of "total time the channel is busy" to "simulation time".

2.2.4.2 Database Statistics

In addition to the file statistics, the Abstract Link MAC model also enters statistics in various scenario statistics database tables. Refer to *QualNet Statistics Database User's Guide* for details.

2.2.4.3 Dynamic Statistics

No dynamic statistics are supported for the Abstract Link MAC model.

2.3 Abstract Satellite Model

2.3.1 Description

This is an abstract model of a satellite network. Each satellite network is grouped into subnets. Each satellite subnet has exactly one satellite node and many ground nodes. The ground nodes associated with a subnet always transmit to the designated subnet satellite node. Thus, no handoffs are involved. Also, satellite nodes are bent-pipe satellites (relay data only). When the satellite node receives data from the ground nodes, it broadcasts the data to all other ground nodes in the subnet, but not to the ground node originating the data. Finally, the satellite node must not be generating any packets. Thus, the satellite node cannot run an application or routing protocol.

Note: This model represents the highest level of abstraction of a satellite model considering of a lossless linear delay process. Developers requiring more detailed simulation of satellite systems should consider using the advanced satellite addon module described in *Satellite Model Library*.

The queuing performance of the system is entirely determined by the network layer processes. Packet loss due to data transmission errors is not modeled. Therefore, the only parameter that needs to be calculated is packet latency which is given by the following equation:

$$t_k = t_{prop} + \frac{l_k}{r}$$

where t_k is the transmission delay of frame k , t_{prop} is the configured propagation delay, l_k is the length of frame k , and r is the configured bandwidth of the satellite communication network.

2.3.2 Command Line Configuration

To enable the Abstract Satellite model, include the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] MAC-PROTOCOL      SATCOM
```

The scope of this parameter declaration can be Global, Node, Subnet, or Interface. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

Abstract Satellite Model Parameters

Table 2-11 shows the Abstract Satellite configuration parameters. See Section 1.2.1.3 for a description of the format used for the parameter table.

TABLE 2-11. Abstract Satellite Model Parameters

| Parameter | Value | Description |
|--|---|---|
| SATCOM-SATELLITE-NODE <i>Required</i> <i>Scope:</i> All | Integer | Specifies which node is the satellite node |
| SATCOM-TYPE <i>Required</i> <i>Scope:</i> All | List: • BENT-PIPE | Specifies the SATCOM type. Currently, only the bent-pipe satellite is supported. |
| SATCOM-BANDWIDTH <i>Optional</i> <i>Scope:</i> All | Integer <i>Range:</i> > 0 <i>Default:</i> 1000000 <i>Unit:</i> bps | Specifies the satellite link bandwidth capacity. |
| SATCOM-PROPAGATION-DELAY <i>Optional</i> <i>Scope:</i> All | Time <i>Range:</i> > 0S <i>Default:</i> 270MS | Specifies the ground-to-ground propagation delay. |
| MAC-LAYER-STATISTICS <i>Optional</i> <i>Scope:</i> All | List: • YES • NO <i>Default:</i> NO | Specifies whether MAC Layer Statistics are enabled or disabled. |

2.3.3 GUI Configuration

This section describes how to configure the Abstract Satellite model in the GUI.

Satellite and Ground Station Devices

Abstract Satellite is modeled by the SATCOM device in the Network Components toolbar of the Standard Toolset.

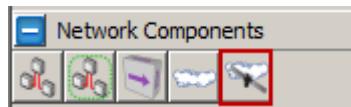


FIGURE 2-7. SATCOM Device in Network Components Toolbar

Ground stations can be modeled by the Default Device or the Ground Station Device in the Devices toolbar of the Standard Toolset.

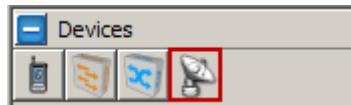


FIGURE 2-8. Ground Station Device in Devices Toolbar

Configuring Abstract Satellite Parameters

To configure the general Abstract Satellite parameters, perform the following steps:

1. Go to **Satellite Properties Editor > Satellite Configuration**.
2. Set the parameters listed in [Table 2-12](#).

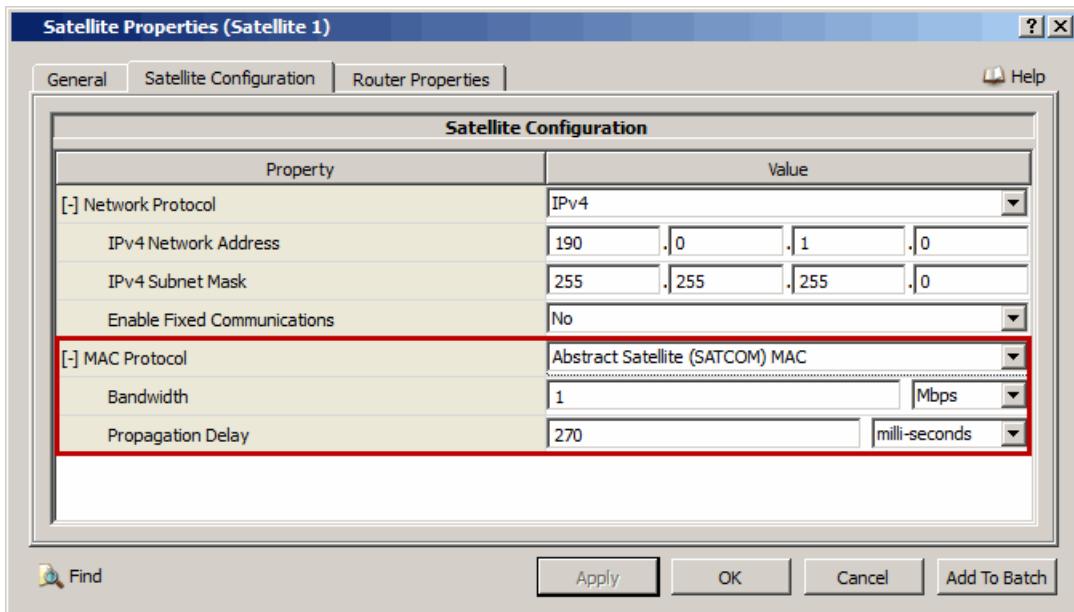


FIGURE 2-9. Setting Abstract Satellite Parameters

TABLE 2-12. Command Line Equivalent of Abstract Satellite Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-------------------|------------------------|--------------------------|
| Bandwidth | Subnet | SATCOM-BANDWIDTH |
| Propagation Delay | Subnet | SATCOM-PROPAGATION-DELAY |

Configuring Statistics Parameters

Statistics for the Abstract Satellite model can be collected at the global, node, subnet and interface levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for Abstract Satellite, check the box labeled **MAC** in the appropriate properties editor.

TABLE 2-13. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|---------------------------------|------------------------|
| MAC | Global, Node, Subnet, Interface | MAC-LAYER-STATISTICS |

2.3.4 Statistics

[Table 2-14](#) lists the Abstract Satellite statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 2-14. Abstract Satellite Statistics

| Statistic | Description |
|-----------------|--|
| Frames sent | Number of frames sent by the SATCOM MAC protocol |
| Frames received | Number of frames received by the SATCOM MAC protocol |
| Frames relayed | Number of frames received by the SATCOM MAC protocol and treated as retransmissions for bent-pipe simulation |

2.3.5 Sample Scenario

2.3.5.1 Scenario Description

The sample scenario consists of a bent-pipe satellite system containing a satellite (node 4) and three ground stations (nodes 1 to 3).

Topology

[Figure 2-10](#) shows the sample scenario topology.

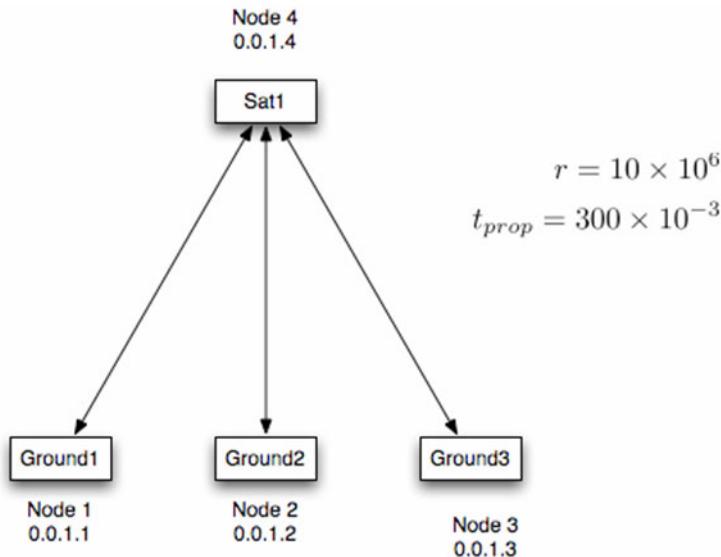


FIGURE 2-10. Sample Abstract Satellite Scenario

2.3.5.2 Command Line Configuration

To configure the sample scenario for the command line, include the following lines in the scenario configuration (.config) file:

```
SUBNET N8-1.0 { 1 thru 4 }
[N8-1.0] MAC-PROTOCOL SATCOM
[N8-1.0] SATCOM-SATELLITE-NODE 4
[N8-1.0] SATCOM-TYPE BENT-PIPE
SATCOM-BANDWIDTH 100000000
SATCOM-PROPAGATION-DELAY 300MS
```

2.3.5.3 GUI Configuration

To configure the sample scenario in the GUI, perform the following steps:

1. Place a SATCOM device and three Ground Station devices on the canvas. Connect the ground stations to the satellite.
2. Go to **Satellite Properties Editor > Satellite Configuration** and set the properties as follows:
 - a. Set **Bandwidth** to *100 Mbps*.
 - b. Set **Propagation Delay** to *300 milli-seconds*.

2.3.6 Scenarios Included in QualNet

The QualNet distribution includes several sample scenarios for the Abstract Satellite model. All scenarios are located in the directory `QUALNET_HOME/scenarios/developer/satcom`. [Table 2-15](#) lists the sub-directory where each scenario is located.

TABLE 2-15. Abstract Satellite Scenarios Included in QualNet

| Scenario | Description |
|----------------------|--|
| one-sat | Shows the operation of a single satellite network. |
| two-sat | Shows the operation of two satellite networks connected by a common ground node. |
| two-sat-shorter-path | Shows the operation of two satellite networks connected by a common ground node. However, the shortest path is through the ground nodes instead of through the satellites. |

2.4 Address Resolution Protocol (ARP)

The QualNet ARP model is based on RFC 826 and RFC 1122.

2.4.1 Description

ARP is a basic protocol used in almost every TCP/IP implementation. When a packet is sent from one host on a LAN to another, the device driver software does not look at the destination Network layer Protocol Address in the packet; rather, it is the MAC layer Hardware Address that determines for which interface the frame is destined. The purpose of this protocol is to translate the logical 32-bit IP address to the corresponding physical address also known as Ethernet address. ARP resides between the Network layer and the MAC layer in the data link layer. It presents a method of converting Network layer Protocol Addresses (such as IP addresses) to MAC layer Hardware Addresses (such as Ethernet addresses). ARP maintains the mapping between the IP address and the MAC address in a table called ARP cache that is essential for the efficient operation of the ARP. The entries in this table are dynamically added and removed. The default life time of dynamic entry is 20 minutes from the time it was created.

ARP is a general protocol, which can be used in any type of broadcast network. The fields in the ARP packet specify the type of the MAC address and the type of the protocol address. ARP is used with most IEEE 802.x LAN media. In particular, it is used with FDDI, Token Ring, and Fast Ethernet, in precisely the same way as it is with Ethernet.

2.4.2 Features and Assumptions

This section describes the implemented features, omitted features, assumptions and limitations of the ARP model.

2.4.2.1 Implemented Features

- All features specified by RFC-826 are implemented.
- Timeout
- Packet buffer
- ARP flooding prevention
- Gratuitous ARP (ARP cache update, when network interface card is changed and the MAC address to its IP address mapping is changed).
- Variable Length Mac addresses support and clear separation between MAC addressees and IP addresses.

2.4.2.2 Omitted Features

- Proxy ARP
- Gratuitous ARP (Duplicate IP address detection).

2.4.2.3 Assumptions and Limitations

- ARP works at the interface between Network and MAC layer.
- Address Resolution support between IPv4 addresses and Ethernet (802.3) MAC addresses.
- ARP will not work for any scenario containing MAC switch.
- 'ARP cache timeout' must be the same for all nodes in a Subnet.
- LLC must be enabled for ARP to work with MAC protocols other than 802.3.
- ARP must be enabled at all nodes of the subnet.

2.4.3 Supplemental Information

ARP is enhanced to deal with situations such as modifications in the network interface card. You can use the QualNet Interface fault configuration to deal with such situations. To specify interface faults, see [Section 11.1](#) for details.

2.4.4 Command Line Configuration

To enable ARP, include the following parameter in the scenario configuration (.config) file.

```
[<Qualifier>] ARP-ENABLED YES
```

The scope of this parameter declaration can be Global, Node, Subnet, or Interface. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

Note: The default value of parameter ARP-ENABLED is NO.

ARP Parameters

[Table 2-16](#) lists the configuration parameters for ARP. See [Section 1.2.1.3](#) for a description of the format used for the parameter tables.

TABLE 2-16. ARP Parameters

| Parameter | Value | Description |
|---|---|--|
| ARP-BUFFER-SIZE <i>Optional</i> Scope: All | Integer <i>Range:</i> ≥ 0 <i>Default:</i> 1 <i>Unit:</i> buffer slot (24 bytes) | This parameter specifies ARP buffer size. |
| ARP-STATIC-CACHE-FILE <i>Optional</i> Scope: Global, Node | Filename | Name of the ARP static cache file. The format of this file is described in Section 2.4.4.2 . |
| MAC-ADDRESS-CONFIG-FILE <i>Optional</i> Scope: All | Filename <i>Default:</i> 1 | Name of the MAC address configuration file. The format of this file is described in Section 2.4.4.1 . |
| ARP-CACHE-EXPIRE-INTERVAL <i>Optional</i> Scope: All | Time <i>Range:</i> > 0 <i>Default:</i> 20M | Timeout interval of the ARP cache table. |
| ARP-STATISTICS <i>Optional</i> Scope: All | List: • YES • NO <i>Default:</i> NO | Indicates whether statistics collection is enabled for ARP. |

Configuring MAC Address in the Scenario Configuration File

MAC addresses can be configured by means of the MAC address configuration file or they can be specified in the scenario configuration (.config). [Table 2-17](#) lists the parameters that can be specified in the scenario configuration file to configure MAC addresses.

TABLE 2-17. Parameters for Configuring MAC Addresses

| Parameter | Value | Description |
|---|---|---|
| MAC-ADDRESS-TYPE <i>Optional</i> Scope: All | List: • ETHERNET • ATM <i>Default:</i> ETHERNET | MAC address type. |
| MAC-ADDRESS-LENGTH <i>Optional</i> Scope: All | Integer <i>Default:</i> 6 <i>Unit:</i> bytes | Length (number of bytes) of the MAC address. |
| MAC-ADDRESS <i>Optional</i> Scope: All | String | MAC address. Note: This parameter must be specified at the interface level. |

2.4.4.1 Format of the MAC Address Configuration File

The MAC address configuration file is used to generate interface MAC Address.

Each line in the MAC address configuration file has the following format:

```
<node ID> <int-index> [<add-length>] [<hardware-type>] <MAC address>
```

where:

| | |
|-----------------|---|
| <node ID> | Node identifier. |
| <int-index> | Interface index. |
| <add-length> | Length of the hardware address. This entry is optional. |
| <hardware-type> | Type of hardware used in the network. The possible values for this field are: <ul style="list-style-type: none">• ETHERNET• ATM This entry is optional. The default hardware type is ETHERNET. |
| <MAC address> | Hardware address. This must be specified in hexadecimal format. |

Example

The following lines show a segment of a MAC address configuration file:

```
1 0 6 ETHERNET 23:4f:5C:aa:FE:B2
2 1 6 ETHERNET 5C-AA-FE-23-4F-C2
4 0 ETHERNET 5C-BA-FE-23-4F-F2
6 0 00:4f:5C:ab:B5:C7
7 0 00:4f:5C:ab:B5:C7
```

2.4.4.2 Format of the ARP Static Cache File

The ARP static cache file is used to create the static ARP Table.

Each line in the ARP static cache file has the following format:

```
<node ID> [int-index] <protocol type> <protocol address> | <node-
interface> <hardware type> <add-length> <hardware address> <timeout>
```

where:

| | |
|--------------------|--|
| <node ID> | Node identifier. The node where the static ARP entry is created. |
| <int-index> | Interface index. This entry is optional. |
| <protocol type> | IP |
| <protocol address> | IP address of the node-interface whose static entry has to be inserted into the ARP cache table. |
| <node-interface> | Node ID & interface of the node whose static entry has to be inserted into the ARP cache table. |
| <hardware-type> | Type of hardware used in the network. It can be either ETHERNET or ATM. |
| <add-length> | Length of the hardware address of the node. |
| <hardware-address> | The 6-byte hardware address of the host whose static entry has been inserted. |
| <timeout> | For static ARP cache table, the time out will be 0 due to forever existence of the entry in the table. |

Note: All fields should be entered on a single line.

Example

The following lines show a segment of an ARP cache file:

```
1 0 IP 3-0 ETHERNET 6 00-00-00-00-03-00 0
2 IP 2-0 ETHERNET 6 00-00-00-00-02-00 0
3 0 IP 192.0.0.1 ETHERNET 6 00-00-00-00-01-00 0
3 1 IP 192.0.1.1 ETHERNET 6 00-00-00-01-01-00 0
```

2.4.5 GUI Configuration

This section describes how to configure ARP in the QualNet GUI.

Configuring ARP Parameters

To configure the ARP parameters, perform the following steps:

1. Go to one of the following locations:
 - To set properties for a specific wireless subnet, go to **Wireless Subnet Properties Editor > Network Layer > ARP**.
 - To set properties for a specific wired subnet, go to **Wired Subnet Properties Editor > ARP**.
 - To set properties for a specific point-to-point link, go to **Point-to-point Link Properties Editor > ARP**.
 - To set properties for a specific node, go to **Default Device Properties Editor > Node Configuration > Network Layer > ARP**.
 - To set properties for a specific interface of a node, go to one of the following locations:
 - **Interface Properties Editor > Interfaces > Interface # > Network Layer > ARP**.
 - **Default Device Properties Editor > Interfaces > Interface # > Network Layer > ARP**.

In this section, we show how to configure ARP parameters for a specific node using the Default Device Properties Editor. Parameters can be set in the other properties editors in a similar way.

2. Set **Enable ARP** to Yes and set the dependent parameters listed in [Table 2-18](#).

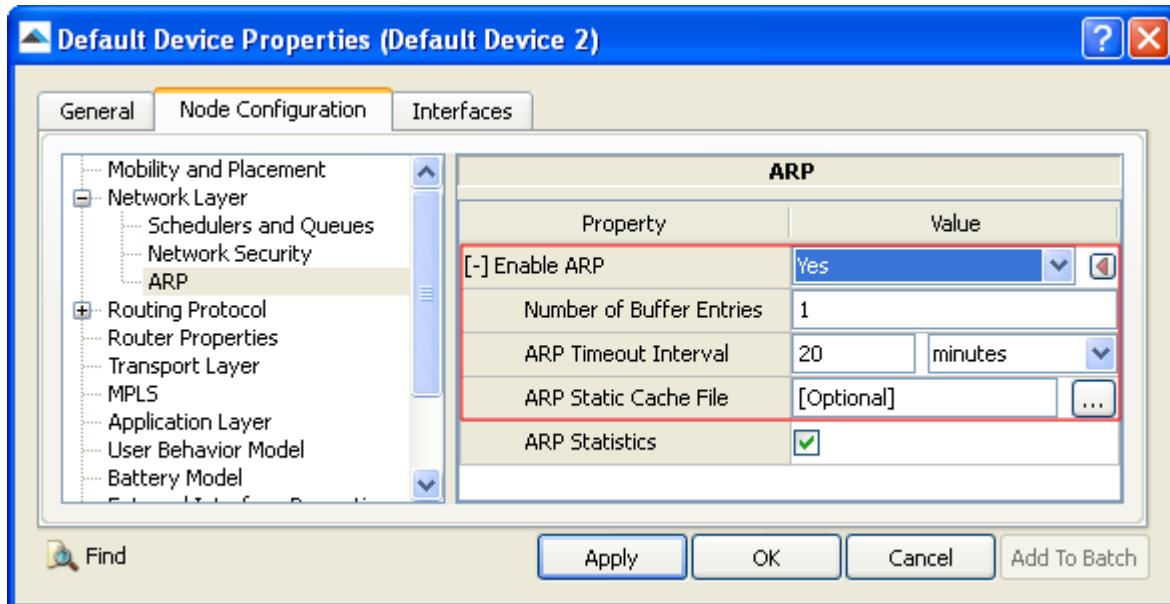


FIGURE 2-11. Setting ARP Parameters

TABLE 2-18. Command Line Equivalent of ARP Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|--------------------------|-------------------------|---------------------------|
| Number of Buffer Entries | Node, Subnet, Interface | ARP-BUFFER-SIZE |
| ARP Timeout Interval | Node, Subnet, Interface | ARP-CACHE-EXPIRE-INTERVAL |
| ARP Static Cache file | Node, Subnet, Interface | ARP-STATIC-CACHE-FILE |

Setting Parameters

- Set **Static Cache File** to the name of the Static Cache file. The format of the Static Cache file is described in [Section 2.4.4.2](#).

Configuring MAC Address

To configure MAC address for an interface, perform the following steps:

1. Go to one of the following locations:
 - **Interface Properties Editor > Interfaces > Interface # > MAC Layer**
 - **Default Device Properties Editor > Interfaces > Interface # > MAC Layer**

2. Set **Configure MAC Address** to Yes and set the dependent parameters listed in [Table 2-19](#).

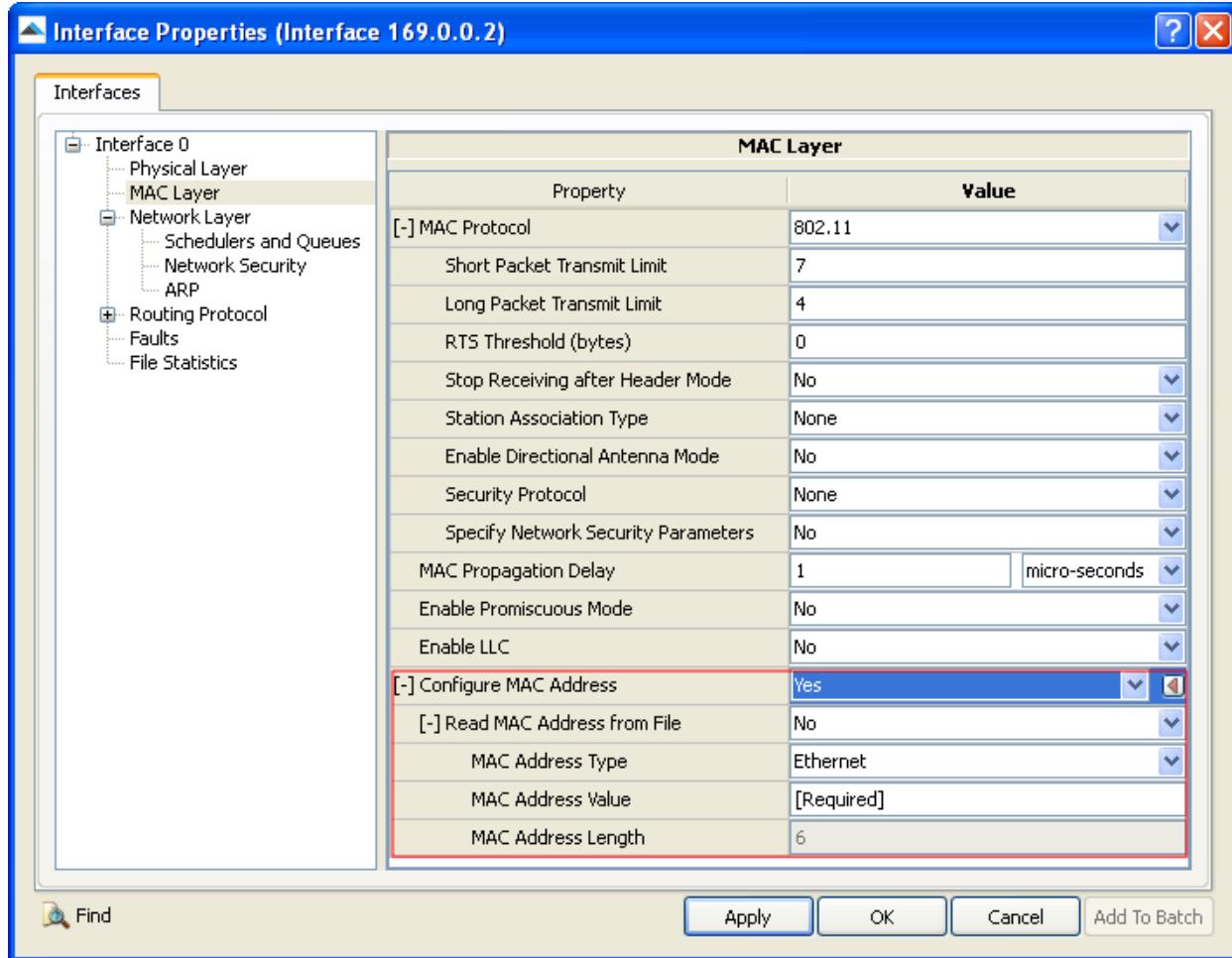


FIGURE 2-12. Setting MAC Address Parameters

TABLE 2-19. Command Line Equivalent of MAC Address Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|----------------------------|------------------------|------------------------|
| Read MAC Address from File | Interface | N/A |
| MAC Address Type | Interface | MAC-ADDRESS-TYPE |
| MAC Address Value | Interface | MAC-ADDRESS |
| MAC Address Length | Interface | MAC-ADDRESS-LENGTH |

3. If **Read MAC Address from File** is Yes, then set the dependent parameters listed in [Table 2-20](#).

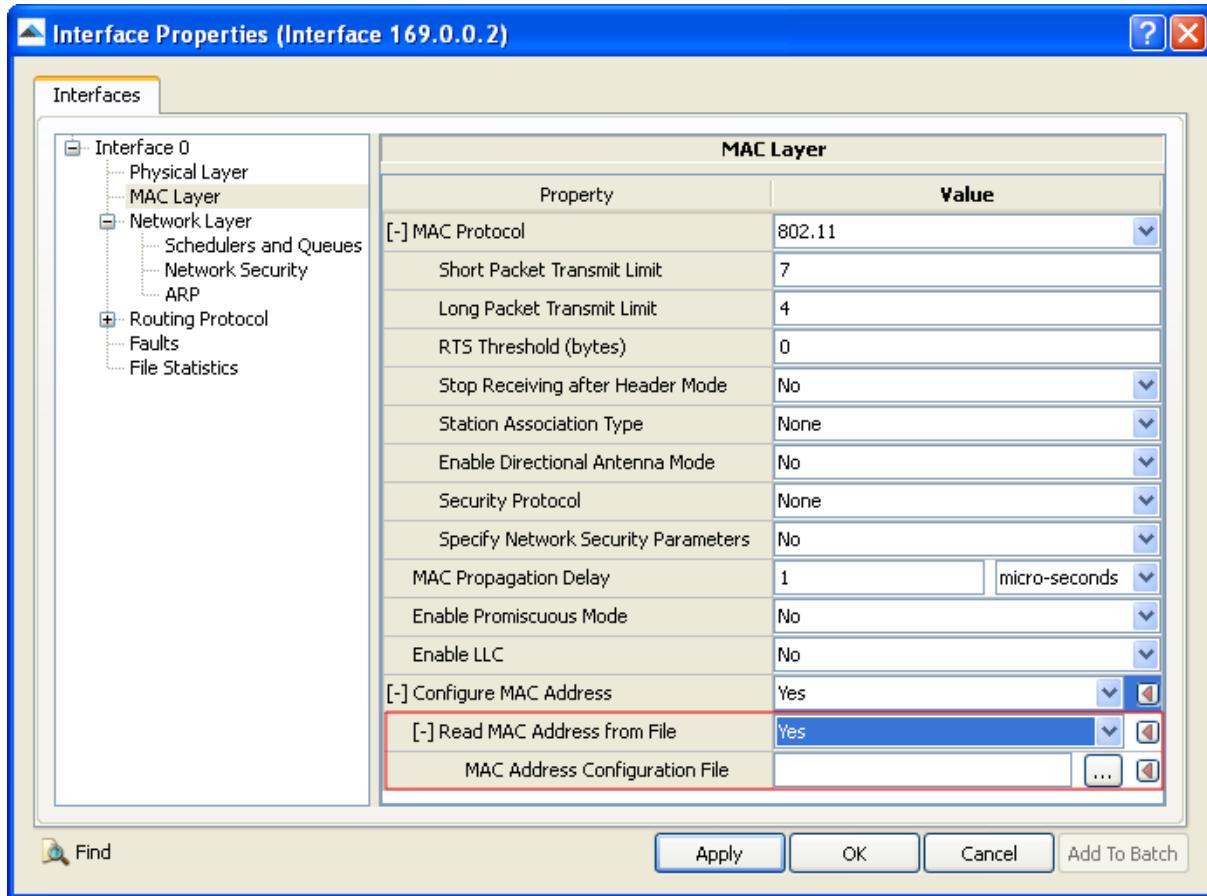


FIGURE 2-13. Setting Enable Hello Processing-specific Parameters

TABLE 2-20. Command Line Equivalent of Enable Hello Processing-specific Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|--------------------------------|------------------------|-------------------------|
| MAC Address Configuration File | Interface | MAC-ADDRESS-CONFIG-FILE |

Setting Parameters

- Set **MAC Address Configuration File** to the name of the MAC Address Configuration File. The format of the MAC Address Configuration File is described in [Section 2.4.4.1](#).

Configuring Statistics Parameters

Statistics for the ARP model can be collected at the global, node, subnet, and interface levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for the ARP model, check the box labeled **ARP Statistics** in the appropriate properties editor.

TABLE 2-21. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|----------------|---------------------------------|------------------------|
| ARP Statistics | Global, Node, Subnet, Interface | ARP-STATISTICS |

2.4.6 Statistics

[Table 2-22](#) lists the ARP statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 2-22. ARP Statistics

| Statistics | Description |
|-------------------------|--|
| Request Sent | Total number of ARP requests sent by the node. |
| Gratuitous Request Sent | Total number of gratuitous ARP requests sent by the node. |
| Request Received | Total number of ARP request packets received by the node. |
| Reply Sent | Total number of ARP reply packets sent by the node. |
| Reply Received | Total number of ARP reply packets received by the node. |
| Data Packet Discarded | Total number of data packets discarded because of inability to resolve MAC address. |
| Cache Entry Inserted | Total number of entries inserted in the ARP cache table. |
| Cache Entry Updated | Total number of entries updated in the ARP cache table. |
| Cache Entry Agedout | Total number of entries expired due to time out. |
| Cache Entry Deleted | Total number of entries deleted when the interface is at fault and the entry has aged out. |

2.4.7 Scenarios Included in QualNet

The QualNet distribution includes several sample scenarios for the ARP model. All scenarios are located in the directory `QUALNET_HOME/scenarios/developer/arp`. [Table 2-23](#) lists the sub-directory where each scenario is located.

TABLE 2-23. ARP Scenarios

| Scenario | Description |
|------------------------------|---|
| 802_3-link-subnet2 | Shows the operation of ARP in the network. |
| 802_3-subnet2 | Shows the operation of ARP in the network. |
| arp-dynamic-routing-protocol | Shows that the routing protocol OSPFv2 works properly with dynamic ARP. |
| arp-static-config-parameter | Shows the possible way to configure static arp. |
| arp-static-routing-protocol | Shows the routing protocol OSPFv2 works properly with static ARP. |

TABLE 2-23. ARP Scenarios (Continued)

| Scenario | Description |
|----------------------------|--|
| autoconfig-subnet1 | Shows the operation of ARP in a single subnet network and checks that the MAC Addresses are generated. |
| buffer-disabled | Shows the operation of ARP in the network with buffer disabled. |
| buffer-enabled | Shows the operation of ARP in the network with buffer enabled. |
| configured-timeout | Shows the operation of ARP in the network and checks that the ARP flushes out-of-date cache entries using user configured Time out values. |
| default-timeout | Shows the operation of ARP in the network and checks that ARP flushes out-of-date cache entries using default Time out values. |
| semiconfig-subnet1 | Shows the operation of ARP in a single subnet network and checks whether user MAC Addresses specifications are working with automatic MAC address specification. |
| usrconfig-subnet1 | Shows the operation of ARP in a single subnet network and checks whether user's MAC Addresses specifications are working. |
| wireless_Arp_timeout_retry | Shows the operation of ARP timeout and ARP retries. |
| wireless_gratuitous | Shows Gratuitous ARP packet generation. |
| wireless_link_subnet | Shows the operation of ARP with two wireless subnets connected via static link. |
| wireless_staticCache | Shows the operation of ARP with ARP static-cache. |
| wireless_subnet1 | Shows the operation of ARP with two wireless subnets connected via common node. |

2.4.8 References

For specifying the interface fault, refer to fault section of Developer Model Library.

1. RFC 826 - An Ethernet Address Resolution Protocol, or Converting Network Protocol Addresses to 48.bit Ethernet Address for Transmission on Ethernet Hardware.
2. TCP/IP Illustrated Volume 1 - Richard Stevens.
3. RFC 1122 - Requirements for Internet Hosts -- Communication Layers

2.5 Logical Link Control (LLC) Protocol

LLC is based on RFC 1042.

2.5.1 Description

Logical Link Control (LLC) is the higher of the two data link layer sub layers defined by the IEEE. The LLC sub layer handles error control, flow control, framing, and MAC-sub layer addressing. The most prevalent LLC protocol is IEEE 802.2, which includes both connectionless and connection-oriented variants.

LLC is a part of the data link layer in a protocol stack. The data link layer controls access to the network medium and defines how upper-layer data in the form of packets or datagram is inserted into frames for delivery on a particular network. The underlying physical layer then transmits the framed data as a stream of bits on the network medium.

2.5.2 Features and Assumptions

This section describes the implemented features, omitted features, assumptions and limitations of the LLC model.

2.5.2.1 Implemented Features

- LLC Type 1 connectionless service.

2.5.2.2 Omitted Features

- XID Command
- Test Command
- LLC Type 2 Service
- LLC Type 3 Service

2.5.2.3 Assumptions and Limitations

None.

2.5.3 Command Line Configuration

To enable LLC, include the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] LLC-ENABLED      YES
```

The scope of this parameter declaration can be Global, Node, Subnet, or Interface. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

- Notes:**
1. To disable LLC, set `LLC-ENABLED` to `NO`. The default value of the parameter `LLC-ENABLED` is `NO`.
 2. When configuring LLC at the node level, make sure that LLC is enabled at both communicating nodes.

LLC Parameters

There are no additional configuration parameters for the LLC model.

2.5.4 GUI Configuration

To configure LLC in the GUI, perform the following steps:

1. Go to one of the following locations:
 - To enable at the subnet level, go to **Wireless Subnet Properties Editor > MAC Layer**.
 - To set properties at the interface level, go to one of the following locations:
 - **Interface Properties Editor > Interfaces > Interface # > MAC Layer**
 - **Default Device Properties Editor > Interfaces > Interface # > MAC Layer**.

In this section, we show how to configure the general 802.11 MAC parameters in the Wireless Subnet Properties Editor. Parameters can be set in the other properties editors in a similar way.

2. Set **Enable LLC** to Yes as shown in [Figure 2-14](#).

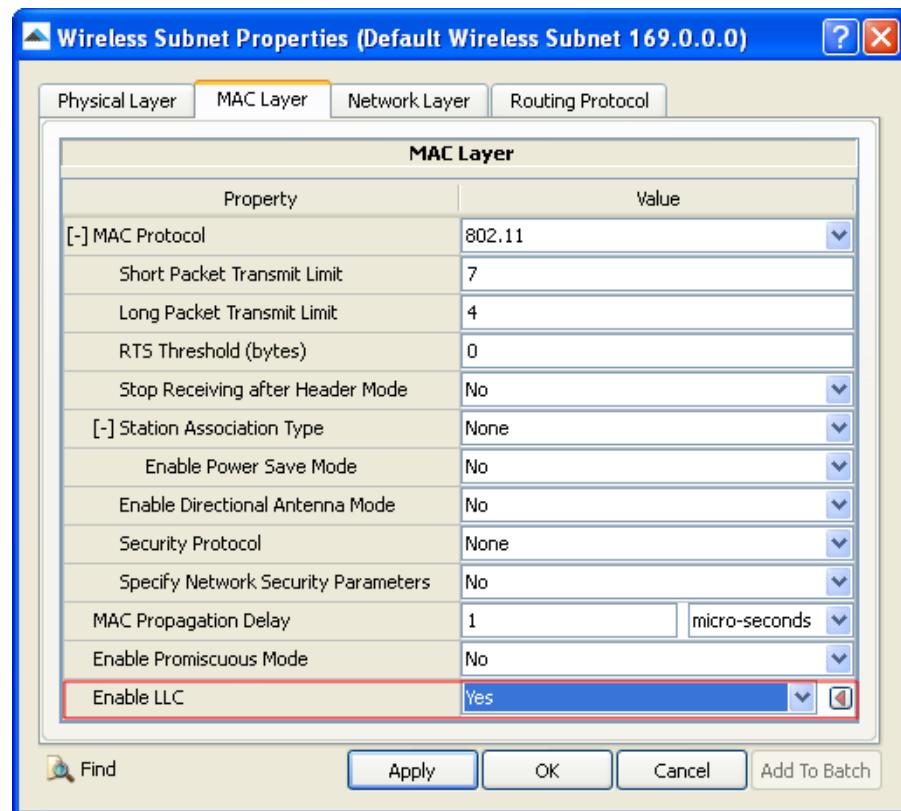


FIGURE 2-14. Enabling LLC

TABLE 2-24. Command Line Equivalent of LLC Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| Enable LLC | Subnet, Interface | LLC-ENABLED |

2.5.5 References

1. RFC 1042, "A Standard for the Transmission of IP Datagrams over IEEE 802 Networks," J. Postel, J. Reynolds, Feb. 1988.
2. CISCO document - Understanding Logical Link Control
http://www.cisco.com/en/US/tech/tk331/tk336/technologies_tech_note09186a0080094777.shtml

3

Network Layer Models

This chapter describes features, configuration requirements and parameters, statistics, and scenarios for Network Layer Models, and consists of the following sections:

- Fixed Communications Model
- Internet Control Message Protocol (ICMP)
- Internet Control Message Protocol version 6 (ICMPv6)
- Internet Group Management Protocol (IGMP)
- Internet Protocol - Dual IP
- Internet Protocol version 4 (IPv4)
- Internet Protocol version 6 (IPv6)
- Neighbor Discovery Protocol

3.1 Fixed Communications Model

3.1.1 Description

The Fixed Communications feature enables the user to enforce a minimum drop rate for application packets and, optionally, a fixed delay for packets that are not dropped. The packets that are not dropped can either completely bypass the simulated network and be delivered at the destination after a specified fixed delay, or they can traverse the simulated network and be subject to network delays.

This feature is particularly useful for cases where the user has computed or has prior information about the delay and drop rate that packets traversing through the target network will be subject to.

3.1.2 Features and Assumptions

This section describes the implemented features, omitted features, assumptions and limitations of the Fixed Communications model.

3.1.2.1 Implemented Features

- Minimum drop probability for application packets
- Fixed delay for packets that are not dropped

3.1.2.2 Omitted Features

None.

3.1.2.3 Assumptions and Limitations

- The model uses constant values for delay and drop probability. Probability distribution functions are not supported.

3.1.3 Command Line Configuration

[Table 3-1](#) lists the configuration parameters for the Fixed Communications model. See [Section 1.2.1.3](#) for a description of the format used for the parameter tables.

TABLE 3-1. Fixed Communication Parameters

| Parameter | Value | Description |
|---|--|--|
| FIXED-COMMS-DROP-PROBABILITY <i>Optional</i> Scope: All | Real <i>Range:</i> [0 . 0 , 1 . 0] <i>Default:</i> 0 . 0 | Probability of dropping application packets. |
| FIXED-COMMS-DELAY <i>Optional</i> Scope: All | Time | Delay after which application packets that are not dropped are delivered at the destination. If this parameter is not specified, the fixed delay feature is not enabled and the packets traverse the simulated network. |

Examples of Parameters Usage

The following are examples of Fixed Communications configuration:

1. Drop 30% of application packets originating from node 2.

```
[2] FIXED-COMMS-DROP-PROBABILITY 0.3
```

2. Application packets sourced from 190.0.1.2 are subject to a deterministic delay of 1 second.

```
[190.0.1.2] FIXED-COMMS-DELAY 1S
```

3. Application packets from all sources are subject to a deterministic delay of 2 seconds.

```
FIXED-COMMS-DELAY 2S
```

3.1.4 GUI Configuration

To configure the Fixed Communications model, perform the following steps:

1. Go to one of the following locations:
 - To set properties for a specific wireless subnet, go to **Wireless Subnet Properties Editor > Network Layer > Fixed Communications**.
 - To set properties for a specific node, go to **Default Device Properties Editor > Node Configuration > Network Layer > Fixed Communications**.
 - To set properties for a specific interface of a node, go to one of the following locations:
 - **Interface Properties Editor > Interfaces > Interface # > Network Layer > Fixed Communications**.
 - **Default Device Properties Editor > Interfaces > Interface # > Network Layer > Fixed Communications**.

In this section, we show how to configure Fixed Communications parameters for a specific node using the Default Device Properties Editor. Parameters can be set in the other properties editors in a similar way.

2. To enable dropping of application packets, set **Packet Drop Probability** to a non-zero value.

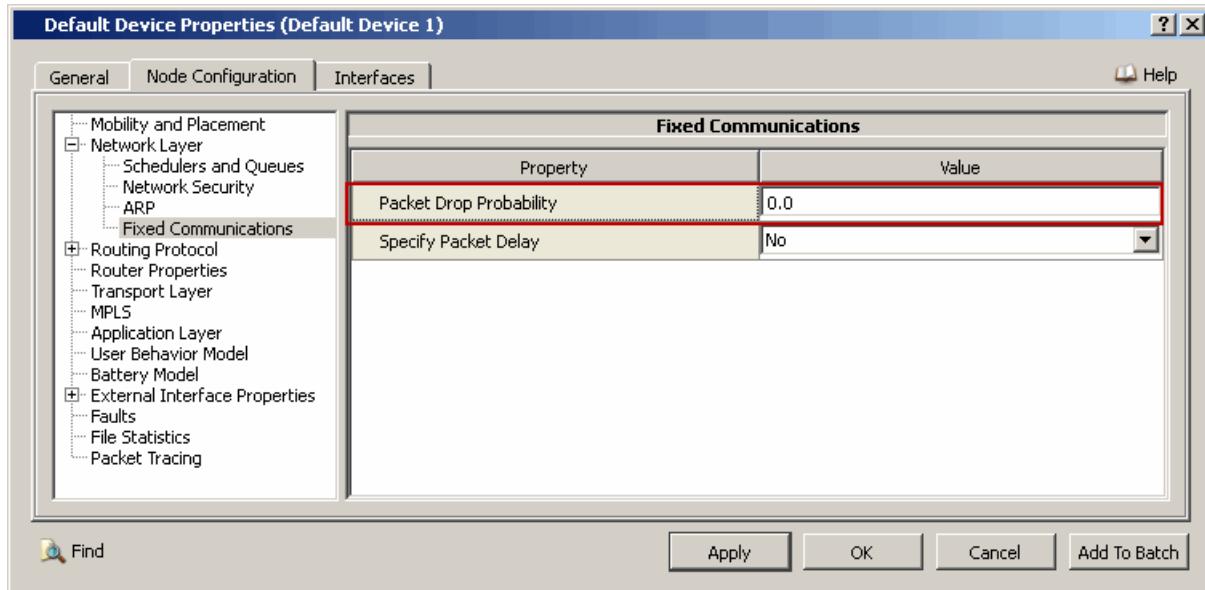


FIGURE 3-1. Specifying Drop Probability

TABLE 3-2. Command Line Equivalent of Drop Probability Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-------------------------|-------------------------|------------------------------|
| Packet Drop Probability | Node, Subnet, Interface | FIXED-COMMS-DROP-PROBABILITY |

3. To specify a fixed packet delay, set **Specify Packet Delay** to Yes and set the dependent parameters listed in [Table 3-3](#).

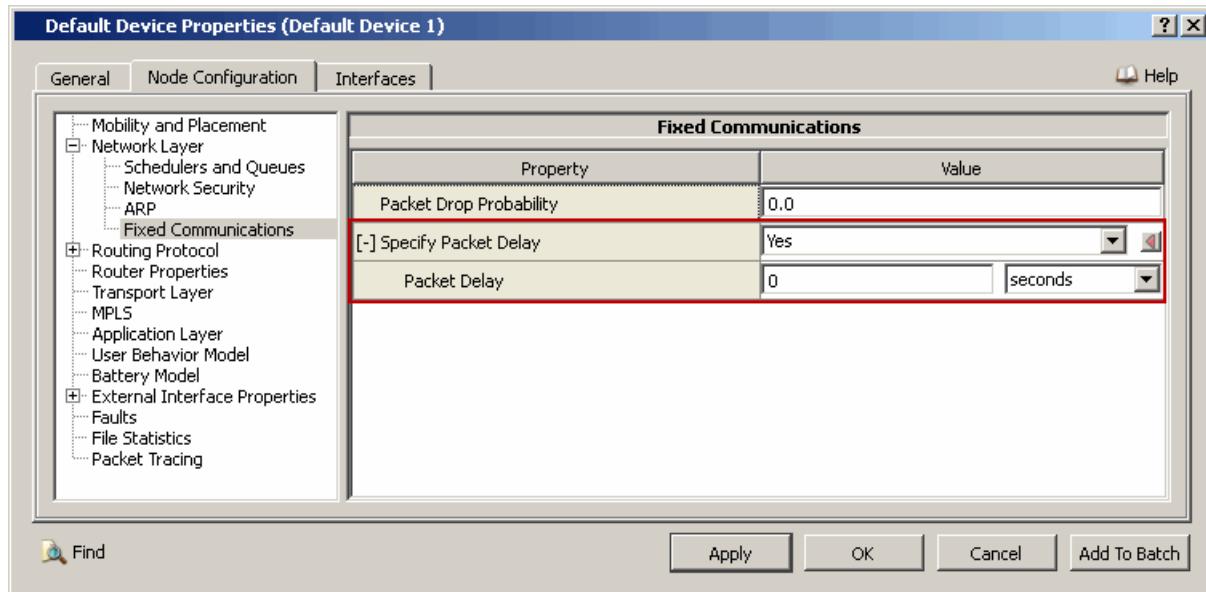


FIGURE 3-2. Specifying Packet Delay

TABLE 3-3. Command Line Equivalent of Packet Delay Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|-------------------------|------------------------|
| Packet Delay | Node, Subnet, Interface | FIXED-COMMS-DELAY |

3.2 Internet Control Message Protocol (ICMP)

The QualNet ICMP model is based on RFC 792, RFC 1122, RFC 1256, RFC 1393, RFC 1349, RFC 1812, and RFC 2521.

3.2.1 Description

ICMP is an integral part of IP. It allows a router or destination host to communicate with the source, to report an error in IP datagram processing and for diagnostic or routing purposes. ICMP messages are constructed at IP layer, from a normal IP datagram that has generated an ICMP response. IP encapsulates the appropriate ICMP message with a new IP header and transmits the resulting datagram in the usual manner.

The Internet Control Message Protocol (ICMP) is one of the core protocols of the Internet Protocol Suite. It is mainly used by networked computers to send error messages indicating, for instance, that a requested service is not available or that a host or router could not be reached. ICMP messages are typically generated in response to errors in IP datagram or for diagnostic or routing purposes.

ICMP relies on IP to perform its tasks, and it is an integral part of IP. It differs in purpose from transport protocols such as TCP and UDP in that it is typically not used to send and receive data between end systems. It is usually not used directly by user network applications, with some notable exceptions being the ping tool and trace route.

ICMP messages are constructed at the IP layer, usually from a normal IP datagram that has generated an ICMP response. IP encapsulates the appropriate ICMP message with a new IP header (to get the ICMP message back to the original sending host) and transmits the resulting datagram in the usual manner. The original sending host, which now becomes the destination for ICMP packet, handles the packet on reception.

3.2.2 Features and Assumptions

3.2.2.1 Implemented Features

- Destination Unreachable
 - Network Unreachable
 - Host Unreachable
 - Protocol Unreachable
 - Port Unreachable
 - Source Route Failed.
 - Fragmentation needed and DF set
 - Handling of Destination Network Unknown
 - Handling of Destination Host Unknown
 - Handling of Network Administratively Prohibited
 - Handling of Host Administratively Prohibited
 - Handling of Network Unreachable for TOS
 - Handling of Host Unreachable for TOS
 - Handling of Communication Administratively Prohibited
 - Handling of Host Precedence Violation

- Handling of Precedence Cutoff in Effect
- Echo Reply
- Source Quench
- Redirect Message
 - Generation and handling of Redirect Datagram for the Host
 - Handling of Redirect Datagram for the Network
 - Handling of Redirect Datagram for the TOS and Network
 - Handling of Redirect Datagram for the TOS and Host
- Time Exceeded
- Parameter Problem
- Timestamp reply
- Trace route
- Photuris, Security failures
 - Authentication Failed

3.2.2.2 Omitted Features

- Destination Unreachable
 - Generation and handling of Destination Network Unknown
 - Generation and handling of Destination Host Unknown
 - Generation and handling of Source Host Isolated
 - Generation and handling of Network Administratively Prohibited
 - Generation and handling of Host Administratively Prohibited
 - Generation and handling of Network Unreachable for TOS
 - Generation and handling of Host Unreachable for TOS
 - Generation and handling of Communication Administratively Prohibited
 - Generation and handling of Host Precedence Violation
 - Generation and handling of Precedence Cutoff in Effect
- Redirect Message
 - Generation of Redirect Datagram for the Network.
 - Generation of Redirect Datagram for the TOS and Network.
 - Generation of Redirect Datagram for the TOS and Host.
- Echo Request.
- Timestamp
- Information Request
- Information Reply
- Address Mask Request
- Address Mask reply
- Photuris, Security failures
 - Bad SPI
 - Decompression Failed
 - Decryption Failed
 - Need Authentication

- Need Authorization
- Type 31 to 41 ICMP messages are not implemented.

3.2.2.3 Assumptions and Limitations

- The following on-demand routing protocols will not work with ICMP when Host Unreachable and Network Unreachable error messages are generated since they use their own routing table instead of the IP forwarding table.
 - Ad-hoc On-demand Distance Vector (AODV) routing protocol
 - Anonymous On-Demand Routing (ANODR) routing protocol
 - DYnamic MANET On-demand (DYMO) routing protocol
 - Dynamic Source Routing (DSR) protocol
 - Landmark Ad-hoc Routing (LANMAR) protocol
 - Inter-zone Routing Protocol (IERP)
- The following MAC protocols will not work with ICMP when Host Unreachable and Network Unreachable error messages are generated since they do not notify the IP layer when packets are dropped.
 - Carrier-Sense MUltiple Access (CSMA) MAC protocol
 - Multiple Access with Collision AVoidance (MACA) MAC protocol
 - Time Division MUltiple Access (TDMA) MAC protocol
 - Aloha MAC protocol
 - GENERICMAC
 - IEEE 802.3 MAC protocol
 - Switched-Ethernet

3.2.3 Command Line Configuration

To enable ICMP, include the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] ICMP YES
```

The scope of this parameter declaration can be Global or Node. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

Note: To disable ICMP, set ICMP to NO. The default value of the parameter ICMP is YES.

ICMP Configuration Parameters

The ICMP configuration parameters are described in [Table 3-4](#). See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 3-4. ICMP Configuration Parameters

| Parameter | Value | Description |
|--|---|--|
| ICMP-ROUTER-LIST Optional <i>Scope: Global, Node</i> | Node-list | List of ICMP routers. If a node is not an ICMP router, then it is an ICMP host. |
| ICMP-REDIRECT-ENABLE Optional <i>Scope: Global, Node</i> | List: • YES • NO <i>Default: NO</i> | Indicates whether ICMP redirect is enabled. |
| ICMP-REDIRECT-RETRY-TIME Optional <i>Scope: Global, Node</i> | Integer <i>Range: ≥ 1S</i> <i>Default: 1S</i> | Indicates how often an ICMP router will attempt to send redirect messages to a node. |
| ICMP-REDIRECT-OVERRIDE-ROUTING Optional <i>Scope: Global, Node</i> | List: • YES • NO <i>Default: YES</i> | If enabled, allows for ICMP redirect messages to override forwarding table entries defined by something other than a static route. |
| ICMP-ROUTER-ADVERTISEMENT-LIFE-TIME Optional <i>Scope: Global, Node</i> | Time <i>Default: 1800S</i> | Life time of ICMP router advertisement. |
| ICMP-ROUTER-ADVERTISEMENT-MIN-INTERVAL Optional <i>Scope: Global, Node</i> | Time <i>Default: 450S</i> | Minimum time interval for ICMP router advertisement. See Note 1. |
| ICMP-ROUTER-ADVERTISEMENT-MAX-INTERVAL Optional <i>Scope: Global, Node</i> | Time <i>Range: > 1S</i> <i>Default: 600S</i> | Maximum time interval for ICMP router advertisement. See Note 2. |
| ICMP-MAX-NUM-SOLICITATION Optional <i>Scope: Global, Node</i> | Integer <i>Range: ≥ 0</i> <i>Default: 3</i> | Maximum number of solicitations. |

TABLE 3-4. ICMP Configuration Parameters

| Parameter | Value | Description |
|--|---|--|
| ICMP-HOST-UNREACHABLE-ENABLED Optional Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Enable Host Unreachable error message generation |
| ICMP-NETWORK-UNREACHABLE-ENABLED Optional Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Enable ICMP Network Unreachable error message generation. |
| ICMP-PROTOCOL-UNREACHABLE-ENABLED Optional Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Enable ICMP Protocol Unreachable error message generation. |
| ICMP-PORT-UNREACHABLE-ENABLED Optional Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Enable ICMP port Unreachable error message generation |
| ICMP-FRAGMENTATION-NEEDED-ENABLED Optional Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Enable ICMP Fragmentation needed error message generation |
| ICMP-SOURCE-QUENCH-ENABLED Optional Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Enable ICMP Source Quench error message generation. |
| ICMP-TTL-EXCEEDED-ENABLED Optional Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Enable TTL Exceed error message generation. |
| ICMP-SOURCE-ROUTE-FAILED-ENABLED Optional Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Enable ICMP Source route Failed error message generation. |

TABLE 3-4. ICMP Configuration Parameters

| Parameter | Value | Description |
|--|---|---|
| ICMP-FRAGMENTS-REASSEMBLY-TIMEOUT-ENABLED Optional Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Enable ICMP Fragment Reassembly time out error message generation. |
| ICMP-PARAMETER-PROBLEM-ENABLED Optional Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Enable ICMP parameter problem error message generation. |
| ICMP-SECURITY-FAILURE-ENABLED Optional Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Enable ICMP security failure error message generation. |
| TRACE-ICMP Optional Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Indicates whether packet tracing is enabled for ICMP. Note: To enable packet tracing, some other parameters need to be configured as well. Refer to Section 4.2.10 of QualNet User's Guide for details. |
| ICMP-ERROR-STATISTICS Optional Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Indicates whether statistics for error messages are collected for ICMP. |
| ICMP-STATISTICS Optional Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Indicates whether statistics are collected for ICMP. |

- Notes:**
1. ICMP-ROUTER-ADVERTISEMENT-MIN-INTERVAL must be less than ICMP-ROUTER-ADVERTISEMENT-MAX-INTERVAL.
 2. ICMP-ROUTER-ADVERTISEMENT-MAX-INTERVAL must be less than ICMP-ROUTER-ADVERTISEMENT-LIFE-TIME.

3.2.4 GUI Configuration

This section describes how to configure ICMP in the GUI.

Configuration Requirements

IPv4 must be enabled to use ICMP. See [Section 3.6](#) for enabling IPv4.

Configuring ICMP Parameters

To configure the ICMP parameters, perform the following steps:

1. Go to **Default Device Properties Editor > Node Configuration > Network Layer**.
2. Set **Enable ICMP** to Yes and set the dependent parameters listed in [Table 3-5](#).

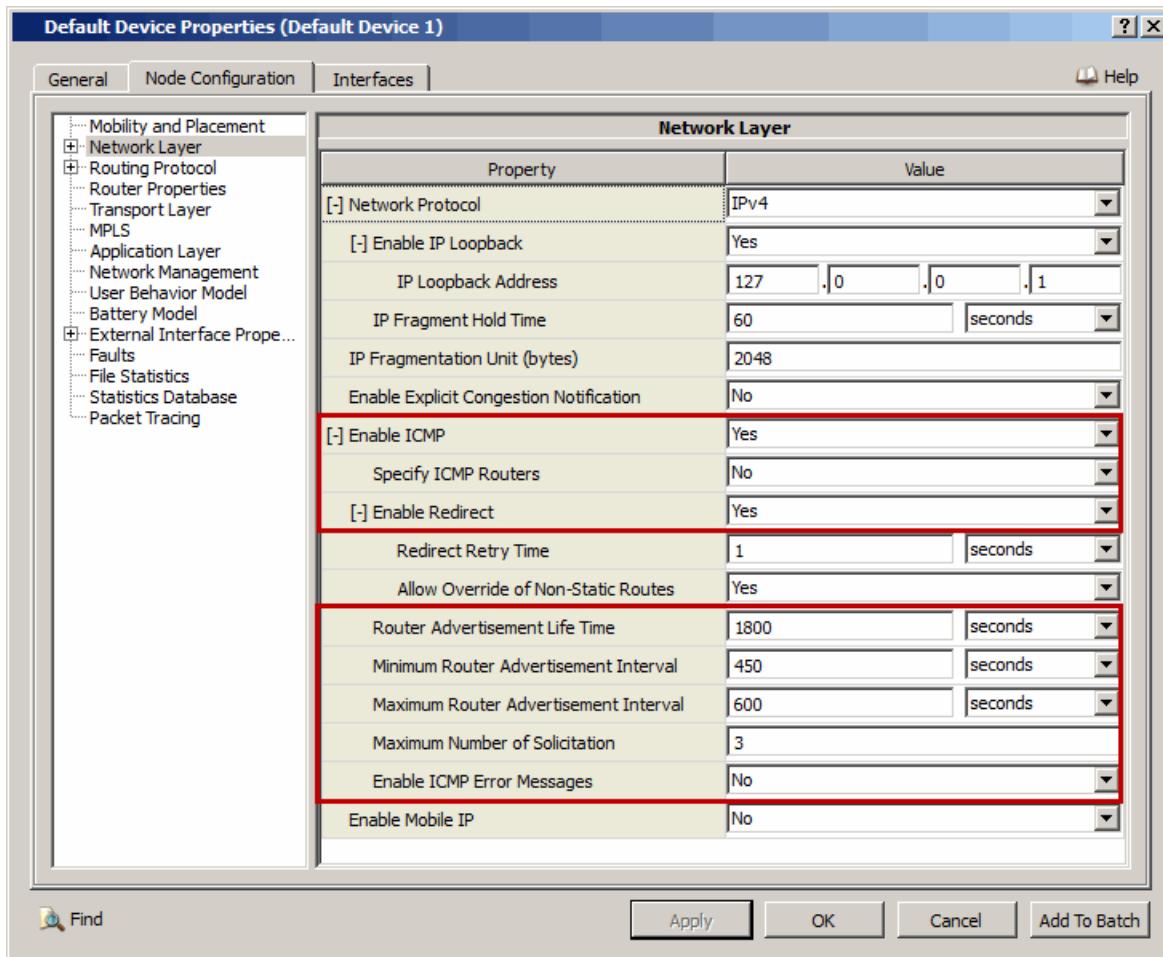


FIGURE 3-3. Setting ICMP Parameters

TABLE 3-5. Command Line Equivalent of ICMP Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------------------------------|-------------------------------|--|
| Specify ICMP Routers | Node | N/A |
| Enable Redirect | Node | ICMP-REDIRECT-ENABLE |
| Router Advertisement Life Time | Node | ICMP-ROUTER-ADVERTISEMENT-LIFE-TIME |
| Minimum Router Advertisement Interval | Node | ICMP-ROUTER-ADVERTISEMENT-MIN-INTERVAL |
| Maximum Router Advertisement Interval | Node | ICMP-ROUTER-ADVERTISEMENT-MAX-INTERVAL |
| Maximum Number of Solicitation | Node | ICMP-MAX-NUM-SOLICITATION |
| Enable ICMP Error Messages | Node | N/A |

Setting Parameters

- To specify the list of ICMP routers, set **Specify ICMP Routers** to Yes; otherwise, set **Specify ICMP Routers** to No.
- To enable ICMP redirect, set **Enable Redirect** to Yes; otherwise, set **Enable Redirect** to No.
- To enable ICMP error messages, set **Enable ICMP Error Messages** to Yes; otherwise, set **Enable ICMP Error Messages** to No.

3. If **Specify ICMP Routers** is set to Yes, then set the parameters listed in [Table 3-7](#).

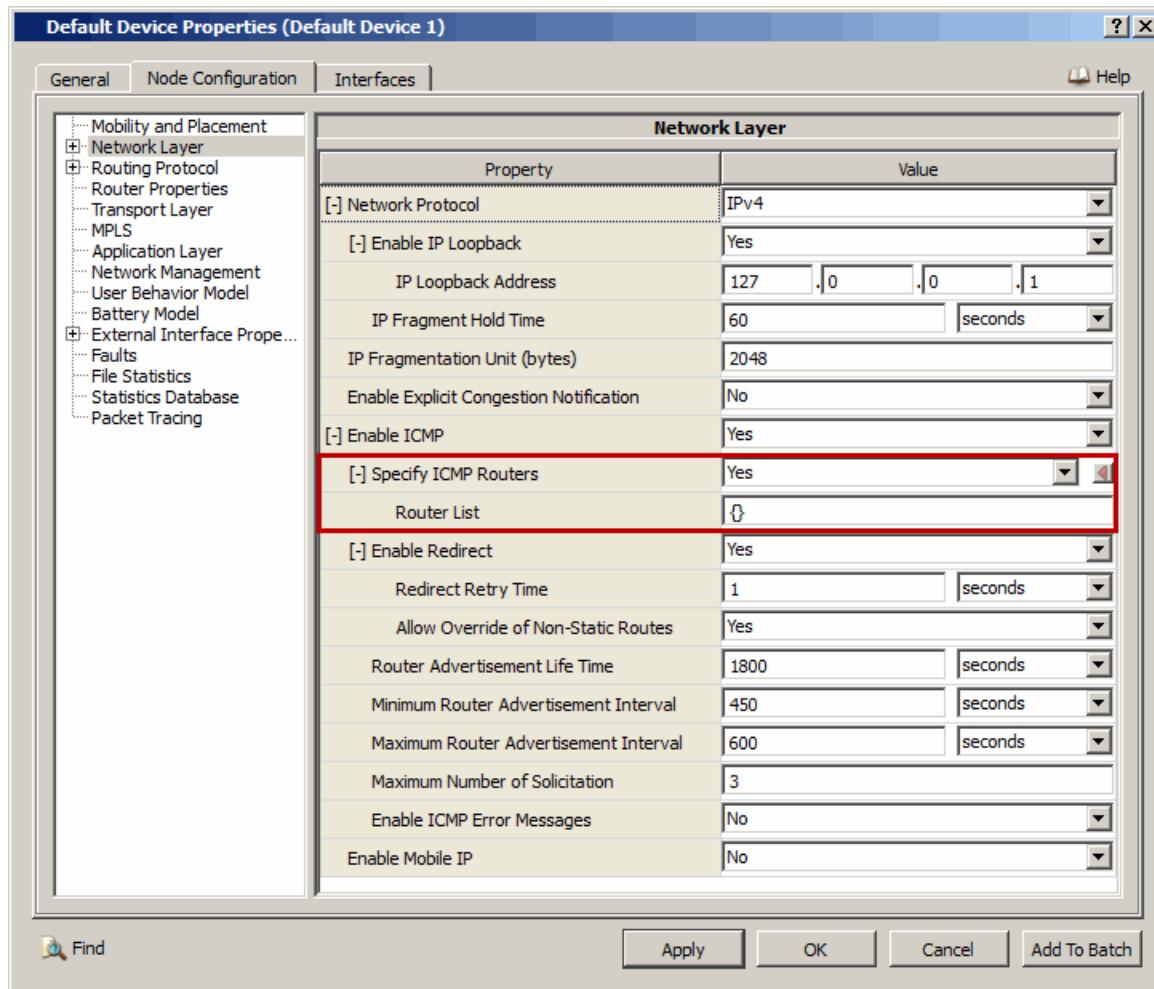


FIGURE 3-4. Setting ICMP Router Parameters

TABLE 3-6. Command Line Equivalent of ICMP Router Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| Router List | Node | ICMP-ROUTER-LIST |

Setting Parameters

- The parameter **Router List** must be set to the same value for all nodes.

4. If **Enable Redirect** is set to Yes, then set the parameters listed in [Table 3-7](#).

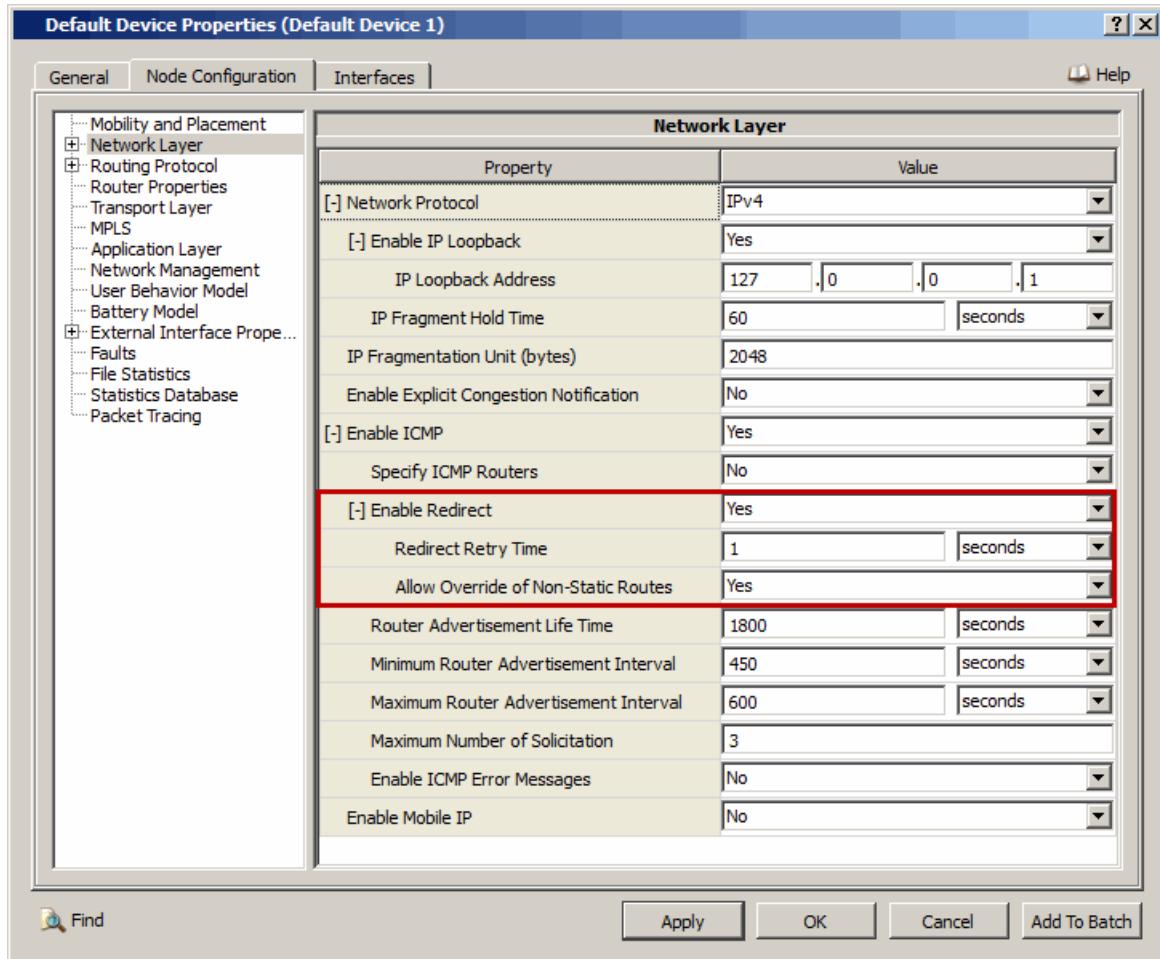


FIGURE 3-5. Setting ICMP Redirect Parameters

TABLE 3-7. Command Line Equivalent of ICMP Redirect Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-------------------------------------|------------------------|--------------------------------|
| Redirect Retry Time | Node | ICMP-REDIRECT-RETRY-TIME |
| Allow Override of Non-static Routes | Node | ICMP-REDIRECT-OVERRIDE-ROUTING |

5. Set **Enable ICMP Error Message** to Yes and set the dependent parameters listed in Table 3-8.

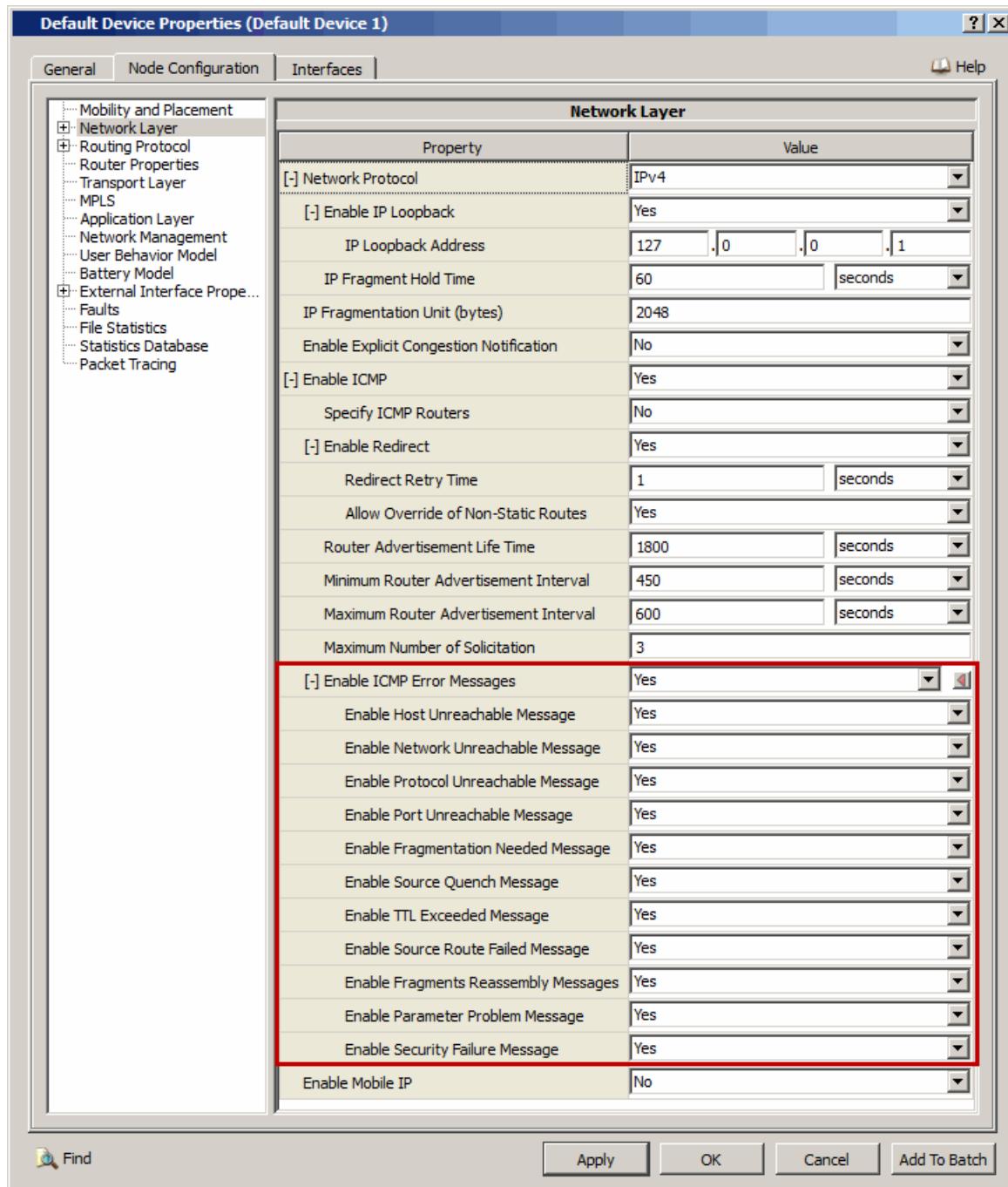


FIGURE 3-6. Setting Error Message Parameters

TABLE 3-8. Command Line Equivalent of ICMP Error Message Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|--------------------------------------|-------------------------------|---|
| Enable Host Unreachable Message | Node | ICMP-HOST-UNREACHABLE-ENABLED |
| Enable Network Unreachable Message | Node | ICMP-NETWORK-UNREACHABLE-ENABLED |
| Enable Protocol Unreachable Message | Node | ICMP-PROTOCOL-UNREACHABLE-ENABLED |
| Enable Port Unreachable Message | Node | ICMP-PORT-UNREACHABLE-ENABLED |
| Enable Fragmentation Needed Message | Node | ICMP-FRAGMENTATION-NEEDED-ENABLED |
| Enable Source Quench Message | Node | ICMP-SOURCE-QUENCE-ENABLED |
| Enable TTL Exceeded Message | Node | ICMP-TTL-EXCEEDED-ENABLED |
| Enable Source Route Failed Message | Node | ICMP-SOURCE-ROUTE-FAILED-ENABLED |
| Enable Fragments Reassembly Messages | Node | ICMP-FRAGMENTS-REASSEMBLY-TIMEOUT-ENABLED |
| Enable Parameter Problem Message | Node | ICMP-PARAMETER-PROBLEM-ENABLED |
| Enable Security Failure Message | Node | ICMP-SECURITY-FAILURE-ENABLED |

Configuring Statistics Parameters

Statistics for ICMP model can be collected at the global and node levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for ICMP model, check the box labeled **ICMP** in the appropriate properties editor.

To enable statistics collection for ICMP Error messages, check the box labeled **ICMP Error Statistics** in the appropriate properties editor.

TABLE 3-9. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-----------------------|-------------------------------|-------------------------------|
| ICMP | Global, Node | ICMP-STATISTICS |
| ICMP Error Statistics | Global, Node | ICMP-ERROR-STATISTICS |

Configuring Packet Tracing Parameters

Packet tracing for ICMP can be enabled at the global and node levels. To enable packet tracing for ICMP, in addition to setting the ICMP trace parameter, *Trace ICMP*, several other trace parameters also need to be set. See Section 4.2.10 of *QualNet User's Guide* for details of configuring packet tracing parameters.

TABLE 3-10. Command Line Equivalent of Packet Tracing Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| Trace ICMP | Global, Node | TRACE - ICMP |

3.2.5 Statistics

Table 3-11 lists the ICMP statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 3-11. ICMP Statistics

| Statistic | Description |
|--|--|
| Advertisements Generated | Specifies the number of advertisements generated. |
| Advertisements Received | Specifies the number of advertisements received. |
| Solicitations Generated | Specifies the number of solicitations generated. |
| Solicitations Received | Specifies the number of solicitations received. |
| Redirect Messages Sent | Specifies the number of redirect messages sent. |
| Redirect Messages Received | Specifies the number of redirect messages received. |
| Echo Messages Received | Specifies the number of echo messages received. |
| Echo Reply Messages Generated | Specifies the number of echo reply messages generated. |
| Timestamp Messages Received | Specifies the number of timestamp messages received. |
| Timestamp Reply Messages Generated | Specifies the number of timestamp reply messages generated. |
| TraceRoute Messages Generated | Specifies the number of trace route messages generated. |
| Network Unreachable Messages sent | Specifies the number of network unreachable messages generated. |
| Network Unreachable Messages Received | Specifies the number of network unreachable messages received. |
| Host Unreachable Messages sent | Specifies the number of host unreachable messages generated. |
| Host Unreachable Messages Received | Specifies the number of host unreachable messages received. |
| Protocol Unreachable Messages sent | Specifies the number of protocol unreachable messages generated. |
| Protocol Unreachable Messages Received | Specifies the number of protocol unreachable messages received. |
| Port Unreachable Messages sent | Specifies the number of port unreachable messages generated. |
| Port Unreachable Messages Received | Specifies the number of port unreachable messages received. |
| Fragmentation Needed Messages sent | Specifies the number of Fragmentation Needed Message generated. |
| Fragmentation Needed Messages Received | Specifies the number of Fragmentation Needed Message send. |
| Source Route Failed Message Sent | Specifies the number of Source Route Failed Message generated. |
| Source Route Failed Message Received | Specifies the number of Source Route Failed Message received. |
| Source Quench Messages sent | Specifies the number of Source quench Message generated. |
| Source Quench Messages Received | Specifies the number of Source quench Message received. |
| TTL Exceeded Messages sent | Specifies the number of TTL Exceed Message generated. |
| TTL Exceeded Messages Received | Specifies the number of TTL Exceed Message received. |
| Fragments Reassembly Messages sent | Specifies the number of Fragment Reassembly Message generated. |

TABLE 3-11. ICMP Statistics (Continued)

| Statistic | Description |
|---|--|
| Fragments Reassembly Messages Received | Specifies the number of Fragment Reassembly Message received. |
| Parameter Problem Messages sent | Specifies the number of Parameter Problem Message generated. |
| Parameter Problem Messages Received | Specifies the number of Parameter Problem Message received. |
| Security Failure Messages sent | Specifies the number of Security Failure Message generated. |
| Security Failure Messages Received | Specifies the number of Security Failure Message received. |
| Destination Network Unknown Messages Received | Specifies the number of Destination Network Unknown Message received. |
| Destination Host Unknown Messages Received | Specifies the number of host Unknown Message received. |
| Network Administratively Prohibited Messages Received | Specifies the number of Network Administratively Prohibited Messages received. |
| Host Administratively Prohibited Messages Received | Specifies the number of Host Administratively Prohibited Messages received. |
| Network Unreachable for TOS Messages Received | Specifies Network Unreachable for TOS Messages Received. |
| Host Unreachable for TOS Messages Received | Specifies Host Unreachable for TOS Messages Received. |
| Communication Administratively Prohibited Messages Received | Specify Communication Administratively Prohibited Messages Received. |
| Host Precedence Violation Messages Received | Specify Host Precedence Violation Messages Received. |
| Precedence Cutoff In Effect Messages Received | Specify Precedence Cutoff In Effect Messages Received. |

3.2.6 Scenarios Included in QualNet

The QualNet distribution includes several sample scenarios for the ICMP model. All scenarios are located in the directory `QUALNET_HOME/scenarios/ developer/icmp`. [Table 3-12](#) lists the sub-directory where each scenario is located.

TABLE 3-12. ICMP Scenarios Included in QualNet

| Scenario | Description |
|--------------------------|---|
| Icmp-host-unreachable | Shows ICMP Host Unreachable Error Message Generation |
| Icmp-netowrk-unreachable | Shows ICMP Network Unreachable Error Message Generation |
| Icmp-security-failure | Shows ICMP Security Failure Error Message Generation |
| Icmp-time-exceed | Shows ICMP TTL-exceed Error Message Generation |
| Icmp-fragmenation-needed | Shows ICMP Fragmentation-Defragmentation Message Generation |
| Icmp-source-quench | Shows ICMP Source Quench Error Message Generation |
| Icmp-redirect | Shows ICMP Redirect Message Generation |

3.2.7 References

1. RFC 792, "Internet Control Message Protocol", J. Postel. September 1981.

2. RFC 1256, "ICMP Router Discovery Messages", S. Deering. September 1991.
3. RFC 1122, "Requirements for Internet Hosts - Communication Layers", R. Braden, October 1989.
4. RFC1812, "Requirements for IP Version 4 Routers", F. Baker, June 1995.
5. RFC 1393, "Trace route Using an IP Option", G. Malkin, January 1993.
6. RFC 1349, "Type of Service in the Internet Protocol Suite", P. Almquist, July 1992.
7. RFC 2521, "ICMP Security Failures Messages", P. Karn, March 1999.

3.3 Internet Control Message Protocol version 6 (ICMPv6)

The QualNet ICMPv6 model is based on RFC 2463.

3.3.1 Description

Internet Control Message Protocol version 6 (ICMPv6) is used by IPv6 nodes to report errors encountered in processing packets, and to perform other internet-layer functions, such as diagnostics (ICMPv6 "ping"). This protocol is an integral part of IPv6. It allows a router or destination host to communicate with the source, typically to report an error in IP datagram processing. ICMPv6 messages are grouped into two classes: error messages and informational messages.

3.3.2 Command Line Configurations

ICMPv6 is automatically enabled if IPv6 is specified as the network protocol. See [Section 3.7](#) for details of configuring IPv6.

ICMPv6 Parameters

The configuration parameter for ICMPv6 is described in [Table 3-13](#). See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 3-13. ICMPv6 Parameters

| Parameter | Value | Description |
|---|---|--|
| NETWORK-LAYER-STATISTICS Optional Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Indicates whether statistics are collected for network protocols. |
| TRACE-ICMPV6 Optional Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Indicates whether packet tracing is enabled for ICMPv6. Note: To enable packet tracing, some other parameters need to be configured as well. Refer to Section 4.2.10 of <i>QualNet User's Guide</i> for details. |

3.3.3 GUI Configuration

ICMPv6 is automatically enabled if IPv6 is specified as the network protocol. See [Section 3.7](#) for details of configuring IPv6.

Configuring Packet Tracing Parameters

Packet tracing for ICMPv6 can be enabled at the global and node levels. To enable packet tracing for ICMP, in addition to setting the ICMP trace parameter, *Trace ICMP*, several other trace parameters also need to be set. See Section 4.2.10 of *QualNet User's Guide* for details of configuring packet tracing parameters.

TABLE 3-14. Command Line Equivalent of Packet Tracing Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| Trace ICMPv6 | Global, Node | TRACE - ICMPV6 |

3.3.4 Statistics

[Table 3-15](#) lists the ICMPv6 statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 3-15. ICMPv6 Statistics

| Statistic | Description |
|---|--|
| Number of Router Advertisement Messages Sent | Number of Router Advertisement Messages Sent. |
| Number of Neighbor Solicitation Messages Sent | Number of Neighbor Solicitation Messages Sent. |
| Number of Neighbor Advertisement Messages Sent | Number of Neighbor Advertisement Messages Sent. |
| Number of Redirect Messages Sent | Number of Redirect Messages Sent. |
| Number of Router Solicitation Messages Received | Number of Router Solicitation Messages Received. |
| Number of Router Advertisement Messages Received | Number of Router Advertisement Messages Received. |
| Number of Neighbor Solicitation Messages Received | Number of Neighbor Solicitation Messages Received. |
| Number of Neighbor Advertisement Messages Received | Number of Neighbor Advertisement Messages Received. |
| Number of Messages Received with Unknown Code | Number of ICMPv6 message whose code field is out of range. |
| Number of Messages Received with Invalid Length | Number of ICMPv6 message whose length field is not matching the actual length. |
| Number of Destination Unreachable Messages Received | Number of Destination Unreachable Messages with code 3 (Address unreachable). |
| Number of Invalid Router Solicitation Messages Received | Number of Bad Router Solicitation Messages Received. |

TABLE 3-15. ICMPv6 Statistics (Continued)

| Statistic | Description |
|--|---|
| Number of Invalid Router Advertisement Messages Received | Number of Bad Router Advertisement Messages Received. |
| Number of Invalid Neighbor Solicitation Messages Received | Number of Bad Neighbor Solicitation Messages Received. |
| Number of Invalid Neighbor Advertisement Messages Received | Number of Bad Neighbor Advertisement Messages Received. |
| Number of Invalid Redirect Messages Received | Number of Bad Redirect Messages Received. |

3.3.5 References

1. RFC 2463, "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6)", A. Conta, S. Deering. December 1998.

3.4 Internet Group Management Protocol (IGMP)

The QualNet IGMP model is based on RFC 2236.

3.4.1 Description

IGMP is used by hosts and routers that support multicasting. IGMP is considered as a part of the IP layer and IGMP messages are transmitted in IP datagrams. Unlike other protocols, IGMP has a fixed size message, with no optional data.

IGMP is a multicast group management protocol. It is generally used between routers and hosts, which involve in multicast data traffic. It operates between a host sending or receiving multicast data and its directly attached router. What we mean by a directly attached router is that router which a host comes across as the first-hop router on the path to any other router outside its local network. Or, on the contrary, the last-hop router on the way to the concerned hosts. The main work of the IGMP enabled routers and hosts are to maintain group membership information over a particular interface on a router on which one or more local hosts are connected. As we can see that the work of IGMP is limited to host and router interaction it becomes very clear that there must be other multicast routing protocols present over the internet to take care of the responsibility of routing multicast data packets to their final destinations. This is accomplished by other network-layer-multicast-routing protocols such as PIM, DVMRP or MOSPF.

We know that IGMP provides a means to the host to inform its directly attached router about its group membership. This is done by the most important three IGMP messages: membership-query (general), membership-query (group-specific) and leave-group. A general-membership-query is sent by a router to all hosts on its attached interface to solicit membership information. By *interface* we mean the primary interface on an attached network. If a router has multiple physical interfaces on a single network, then this protocol needs to run on only one of them (according to RFC 2236). That is to get the set of all the multicast-groups that may have been joined by one or more hosts on that interface. The routers can also determine whether a specific group has been joined by any particular host or not. The routers may query with a specific group address in the query message, to know whether any host on the interface has joined the group or not. The attached hosts on the interface in turn replies with membership-report. This carries the membership information to the querying router. This continues till there are group members on a particular interface. If there are no more reports received for a group on a particular interface, then the attached router understands that there are no more alive group members for a particular group on a particular interface. It then deletes the group information from the group list it carries with it. Thus the router no more forwards multicast data traffic for that particular group on that interface.

IGMP Proxy

IGMP Proxy provides a means for forwarding multicast packets without having to run a multicast protocol. This can be done only for certain simple topologies. An IGMP Proxy device can gather group membership information and simply forward multicast packets based upon that information. In addition, the multicast destination node need not be on the same subnet as the multicast router.

The IGMP Proxy topology is a simple tree with the root connected to a wider multicast infrastructure. Each IGMP Proxy device has one upstream interface (the one towards the root) and can have one or more downstream interface(s). The upstream interface is also called the Host interface and a downstream interface is also called a Router interface. The downstream interface is connected to either a multicast destination node or another IGMP Proxy device. The upstream interface is connected to either another IGMP Proxy device or the multicast router. The upstream interface of an IGMP Proxy device must be manually configured.

3.4.2 Features and Assumptions

This section describes the implemented features, omitted features, assumptions and limitations of the IGMP model.

3.4.2.1 Implemented Features

The model implements the following IGMP Proxy features:

- Maintenance of subscription set at each downstream interface and a merged set of subscription at upstream interface.
- Packet forwarding on downstream interfaces.

3.4.2.2 Omitted Features

None.

3.4.2.3 Assumptions and Limitations

The IGMP model is not compatible with IGMPv1.

3.4.3 Command Line Configuration

To select IGMP as the group management protocol, include the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] GROUP-MANAGEMENT-PROTOCOL    IGMP
```

The scope of this parameter declaration can be Global, Node, Subnet, or Interface. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

IGMP Parameters

[Table 3-16](#) describes the IGMP configuration parameters. [Table 3-17](#) describes the parameters for configuring the IGMP-specific tables in the statistics database tables (refer to *QualNet Statistics Database User's Guide* for details).

See [Section 1.2.1.3](#) for a description of the format used for the parameter tables.

TABLE 3-16. IGMP Parameters

| Parameter | Value | Description |
|---|--|--|
| IGMP-ROUTER-LIST <i>Optional</i> Scope: All | List of nodes and interfaces (see description) | <p>List of nodes and interfaces that will act as IGMP routers.</p> <p>The router list is specified as a list of node IDs and interface addresses separated by commas and enclosed in { and }.</p> <p>Example:</p> <pre>{1, 2, 190.0.0.1, 168.10.12.03}</pre> <p>Note: All other nodes and interfaces that are not in this IGMP router list are considered IGMP hosts. If a node is included in this list, all its interfaces act as IGMP routers.</p> |
| IGMP-PROXY <i>Optional</i> Scope: Global, Node | List: • YES • NO <i>Default:</i> NO | Configures the node as an IGMP Proxy device. |
| IGMP-PROXY-UPSTREAM-INTERFACE <i>Optional</i> Scope: Global, Node | IP Address | <p>IP address of the upstream interface of the IGMP Proxy device.</p> <p>This parameter is required if <code>IGMP-PROXY</code> is set to YES.</p> |
| IGMP-UNSOLICITED-REPORT-COUNT <i>Optional</i> Scope: All | Integer <i>Range:</i> > 0 <i>Default:</i> 2 | Number of times a host will send unsolicited reports at the time of joining a group. |
| IGMP-ROBUSTNESS-VARIABLE <i>Optional</i> Scope: All | Integer <i>Range:</i> > 0 <i>Default:</i> 2 | <p>Robustness variable.</p> <p>The robustness variable allows tuning for the expected packet loss on a subnet. If a subnet is expected to be lossy, the robustness variable may be increased.</p> <p>IGMP is robust to (robustness variable - 1) packet losses.</p> |
| IGMP-QUERY-INTERVAL <i>Optional</i> Scope: All | Time <i>Range:</i> > 0S <i>Default:</i> 125S | Interval between General Query messages sent. |
| IGMP-QUERY-RESPONSE-INTERVAL <i>Optional</i> Scope: All | Time <i>Range:</i> > 0S <i>Default:</i> 10S | Maximum amount of time that the IGMP router waits to receive a response to a General Query message. |

TABLE 3-16. IGMP Parameters (Continued)

| Parameter | Value | Description |
|---|---|---|
| IGMP-LAST-MEMBER-QUERY-RESPONSE-INTERVAL <i>Optional</i> Scope: All | Time <i>Range:</i> (0S, 25.5S) <i>Default:</i> 1S | Amount of time that the IGMP router waits to receive a response to a Group-Specific Query message. This is also the amount of time between successive Group-Specific Query messages. |
| IGMP-STATISTICS <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Indicates whether statistics are collected for the IGMP protocol. |

Table 3-17 lists the parameters for configuring IGMP-specific tables in the statistics database.

TABLE 3-17. IGMP Statistics Database Tables Configuration Parameters

| Parameter | Value | Description |
|---|---|--|
| STATS-DB-MULTICAST-IGMP-SUMMARY-TABLE <i>Optional</i> Scope: Global | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Indicates whether the IGMP Summary table is to be generated. The time between consecutive entries in the IGMP Summary table is determined by the parameter STATS-DB-SUMMARY-INTERVAL. |

3.4.4 GUI Configuration

This section describes how to configure IGMP in the GUI.

Configuring IGMP Parameters

To configure the IGMP parameters, perform the following steps:

1. Go to one of the following locations:
 - To set properties for a specific wireless subnet, go to **Wireless Subnet Properties Editor > Routing Protocol > General**.
 - To set properties for a specific wired subnet, go to **Wired Subnet Properties Editor > Routing Protocol > General**.
 - To set properties for a specific point-to-point link, go to **Point-to-point Link Properties Editor > Point-to-point Link Properties > Routing Protocol**.
 - To set properties for a specific node, go to **Default Device Properties Editor > Node Configuration > Routing Protocol**.
 - To set properties for a specific interface of a node, go to one of the following locations:
 - **Interface Properties Editor > Interfaces > Interface # > Routing Protocol**
 - **Default Device Properties Editor > Interfaces > Interface # > Routing Protocol**.

In this section, we show how to configure IGMP parameters for a specific node using the Default Device Properties Editor. Parameters can be set in the other properties editors in a similar way.

2. Set **Enable Multicast** to Yes and set the dependent parameters listed in [Table 3-18](#).

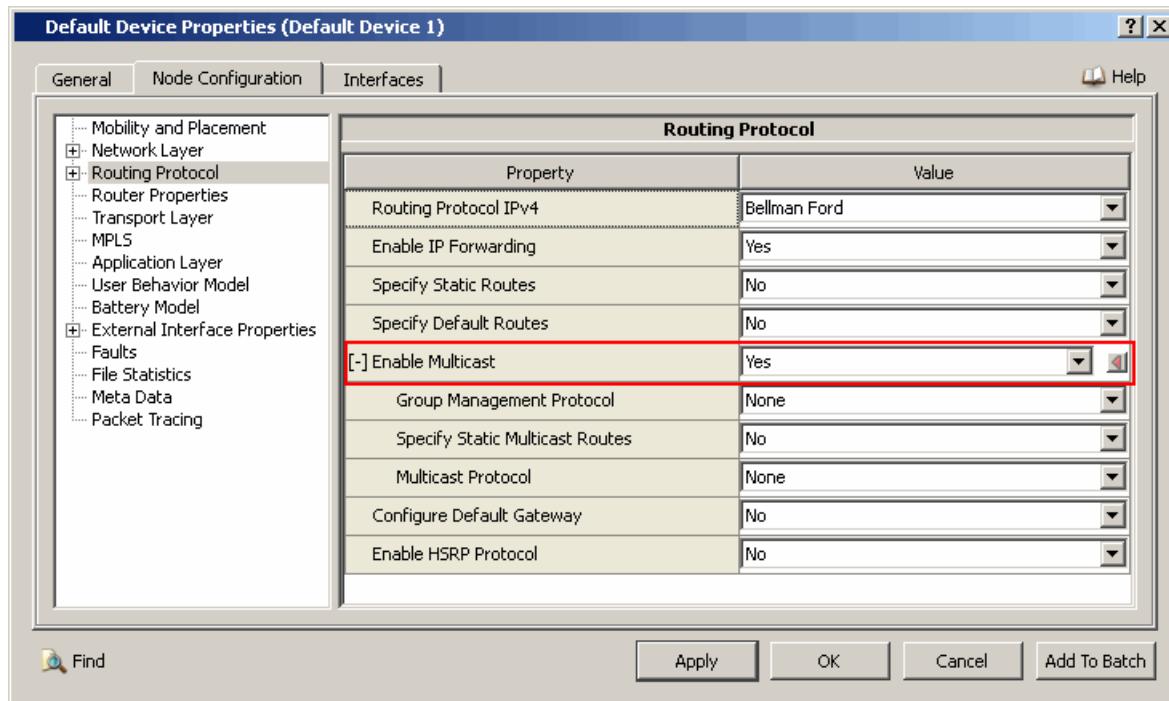


FIGURE 3-7. Enabling Multicast

TABLE 3-18. Command Line Equivalent of Group Management Protocol Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------------------|-------------------------|---------------------------|
| Group Management Protocol | Node, Subnet, Interface | GROUP-MANAGEMENT-PROTOCOL |

Setting Parameters

- Set **Group Management Protocol** to */GMP* to enable IGMP.

3. If **Group Management Protocol** is set to *IGMP*, then set the dependent parameters listed in Table 3-19.

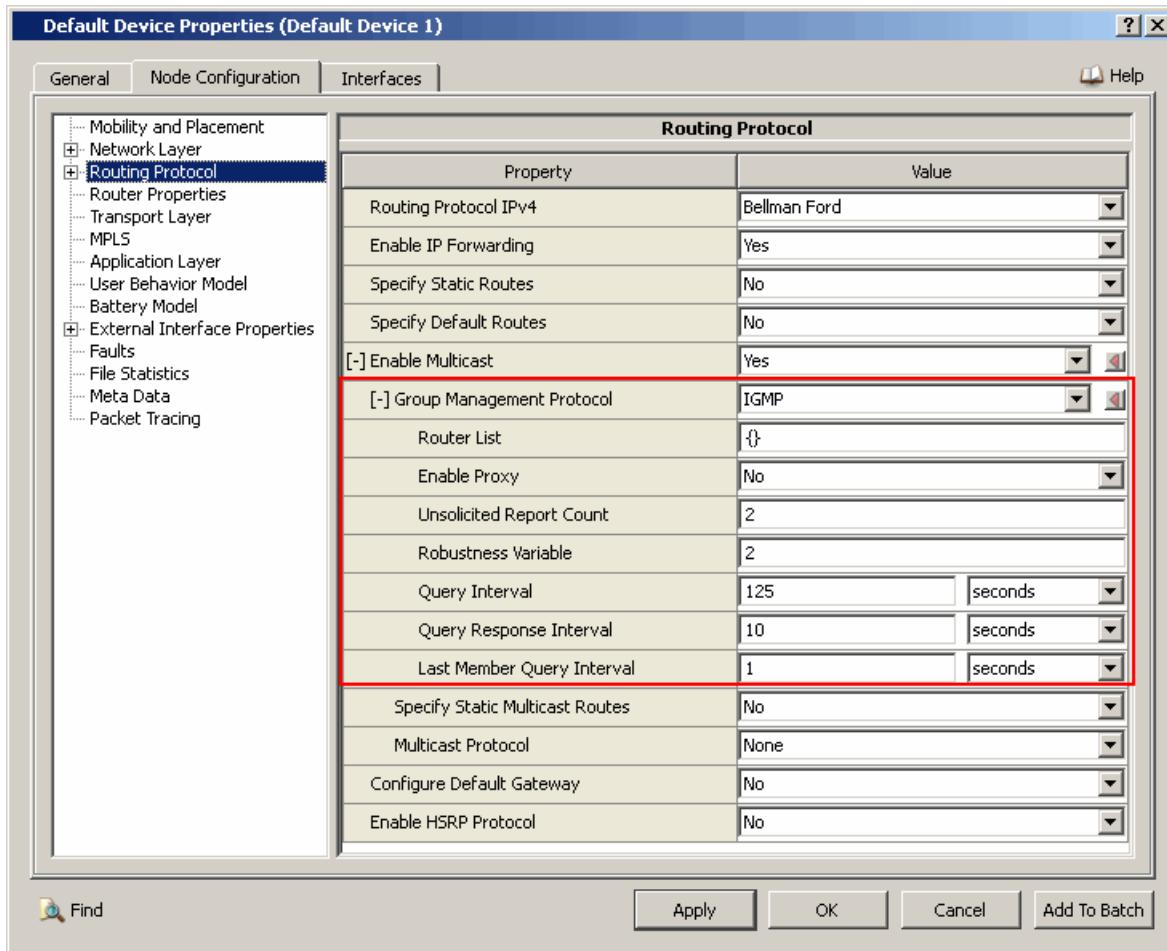


FIGURE 3-8. Setting IGMP Parameters

TABLE 3-19. Command Line Equivalent of IGMP Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|----------------------------|-------------------------|---------------------------------|
| Router List | Node, Subnet, Interface | IGMP-ROUTER-LIST |
| Enable Proxy | Node | IGMP-PROXY |
| Unsolicited Report Count | Node, Subnet, Interface | IGMP-UNSOLICITED-REPORT-COUNT |
| Robustness Variable | Node, Subnet, Interface | IGMP-ROBUSTNESS-VARIABLE |
| Query Interval | Node, Subnet, Interface | IGMP-QUERY-INTERVAL |
| Query Response Interval | Node, Subnet, Interface | IGMP-QUERY-RESPONSE-INTERVAL |
| Last Member Query Interval | Node, Subnet, Interface | IGMP-LAST-MEMBER-QUERY-INTERVAL |

Setting Parameters

- Set **Router List** to the IGMP router list.
The router list is specified as a list of node IDs and interface addresses separated by commas and enclosed in { and }, e.g., {1, 2, 190.0.0.1, 168.10.12.03}.
- Set **Enable Proxy** to Yes to configure the node as an IGMP Proxy device.

4. If **Enable Proxy** is set to Yes, then set the dependent parameters listed in [Table 3-20](#).

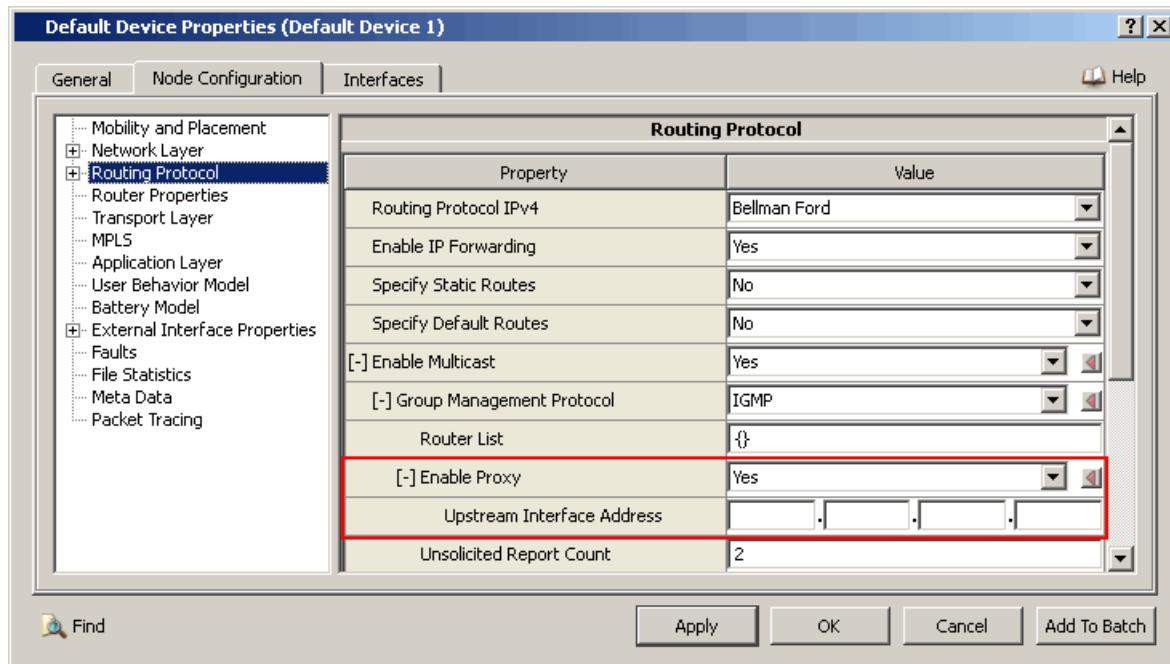


FIGURE 3-9. Setting IGMP Proxy Parameters

TABLE 3-20. Command Line Equivalent of IGMP Proxy Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|----------------------------|------------------------|-------------------------------|
| Upstream Interface Address | Node | IGMP-PROXY-UPSTREAM-INTERFACE |

Configuring File Statistics Parameters

File statistics for IGMP can be collected at the global and node levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

TABLE 3-21. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| IGMP | Global, Node | IGMP-STATISTICS |

Configuring Statistics Database Parameters

To configure the IGMP-specific tables in the statistics database, perform the following steps:

1. Go to **Scenario Properties Editor > Statistics > Statistics Database**.
2. Set **Enable Statistics Database** to Yes.
3. Set **Model-specific Tables** set to Yes and set the IGMP database table parameters listed in [Table 3-22](#).

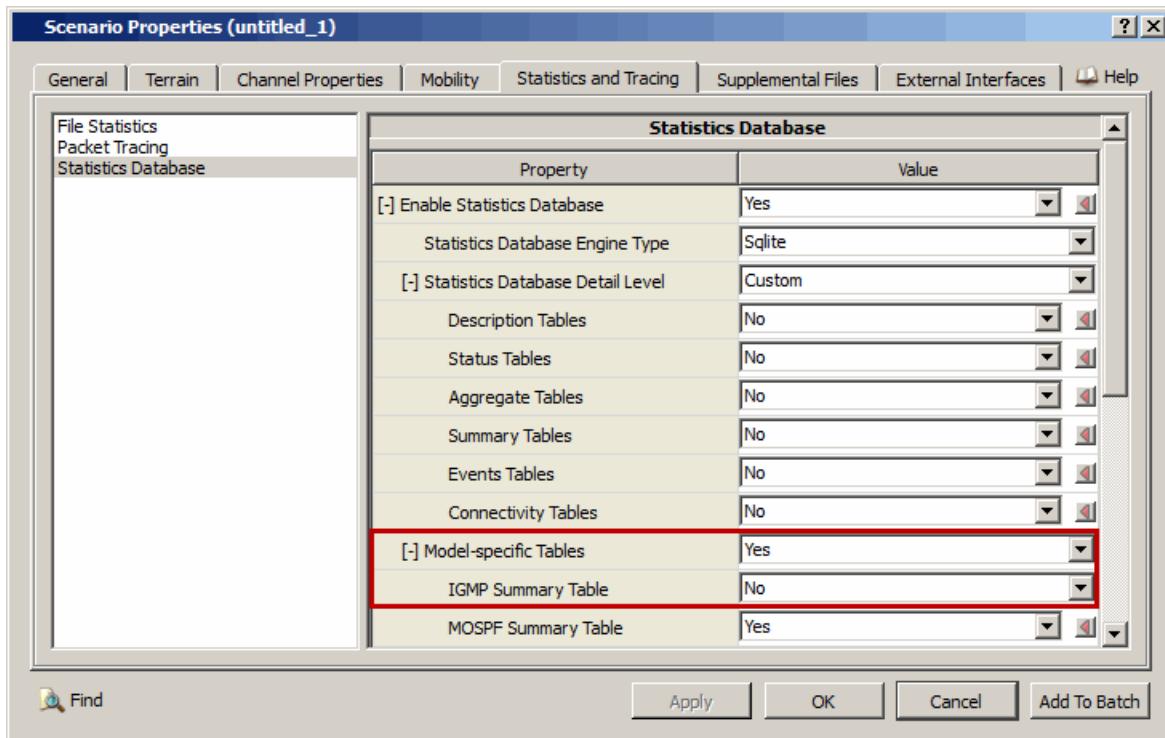


FIGURE 3-10. Configuring IGMP Tables in Statistics Database

TABLE 3-22. Command Line Equivalent of IGMP Statistics Database Table Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|--------------------|------------------------|---|
| IGMP Summary Table | Global | STATS - DB - MULTICAST - IGMP - SUMMARY - TABLE |

3.4.5 Statistics

This section describes the file, database, and dynamic statistics of the IGMP model.

3.4.5.1 File Statistics

[Table 3-23](#) lists the IGMP statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 3-23. IGMP Statistics

| Statistic | Description |
|-------------------------------------|--|
| Membership created in local network | Total number of memberships created in the local network |
| Total Group Joins | In case of host it is the number of groups the host joined through this interface, while in case of router it is the total number of different groups for which the router receives membership reports through this interface. |
| Total Group Leaves | In case of host this variable indicates whether the host leaves a previously joined group through this interface, while for router it is the total number of groups that leave this network. |
| Total Messages Sent | Total number of messages sent. |
| Total Messages Received | Total number of messages received. |
| Bad Messages Received | Total number of bad messages received. |
| Total Reports Received | Total number of reports received. |
| Bad Reports Received | Total number of invalid reports received. |
| Membership Reports Received | Total number of membership reports received. |
| Leave Reports Received | Total number of leave reports received. |
| Total Queries Received | Total number of queries received. |
| Bad Queries Received | Total number of invalid queries received. |
| General Queries Received | Total number of general queries received. |
| Group Specific Queries Received | Total number of group specific queries received. |
| Total Reports Sent | Total number of reports sent. |
| Membership Reports Sent | Total number of membership reports sent. |
| Leave Reports Sent | Total number of leave reports sent. |
| Total Queries Sent | Total number of queries sent. |
| General Queries Sent | Total number of general queries sent. |
| Group Specific Queries Sent | Total number of group specific queries sent. |

3.4.5.2 Database Statistics

In addition to the file statistics, the IGMP model also enters statistics in various scenario statistics database tables. Refer to *QualNet Statistics Database User's Guide* for details.

The IGMP model also enters statistics in the following IGMP-specific database table:

- IGMP Summary Table

3.4.5.3 Dynamic Statistics

No dynamic statistics are supported for the IGMP model.

3.4.6 Scenarios Included in QualNet

The QualNet distribution includes several sample scenarios for the IGMP model. All scenarios are located in the directory `QUALNET_HOME/scenarios/developer/igmp`. [Table 3-24](#) lists the sub-directory where each scenario is located.

TABLE 3-24. IGMP Scenarios Included in QualNet

| Scenario | Description |
|--------------------|--|
| grp-mem-info-db | Shows the generation of multicast group information on all interfaces of a Router. |
| grp-mem-rpt | Shows sending of Group Membership Report Message by Hosts of specific groups in response to General Query. |
| mcast-grp-addr | Shows Group Address field of IGMP messages. |
| qr-rtr-cnt | Shows the number of Querier Router in a single subnet is 1. |
| rtr-host-join-grps | Shows the generation of proper group information on an interface of a Router (both querier and non querier) when both Router and Hosts join different multicast groups in a mixed scenario. |
| rtr-join-grp | Shows the router behavior similar to host with respect to sending unsolicited reports upon joining a group and responding to queries transmitted by itself. |
| sel-qr-rtr | Shows the selection of Querier among the routers starting as querier Router. |
| trans-qr-to-non-qr | Shows the transition from querier to non querier should not occur if that very querier receives a leave report. |
| tx-leave-grp-mess | Shows the behavior of a host of a particular group which transmits a Leave Group Message while it is the last member of the same to respond to the queries which are sent by the querier router. |

3.4.7 References

1. RCC 2236, “Internet Group Management Protocol, Version 2”, W. Fenner. November 1997.
2. RFC 4605, “Internet Group Management Protocol (IGMP) / Multicast Listener Discovery (MLD)-Based Multicast Forwarding”.

3.5 Internet Protocol - Dual IP

The QualNet Dual IP model is based on the RFC 4213, RFC 2185 and RFC 3056.

3.5.1 Description

IPv6 transition might become successful if IPv4/IPv6 routers/hosts maintain compatibility with the large installed base of IPv4 hosts and routers. Maintaining compatibility with IPv4 means enabling IPv6 routers/hosts to carry IPv4/IPv6 datagram over the attached IPv4 infrastructures by encapsulating the datagram within IPv4 datagram. The harmonious co-existence of IPv6 and IPv4 routers/hosts facilitates deployment of IPv6 internet among the large installed base of IPv4. IPv4 compatibility mechanisms include:

- Dual IP Layer: For supporting both IPv6 and IPv4 functionalities.
- Configured tunneling of IPv6 over IPv4: The IPv4 tunnel endpoint address is determined by configuration information on the encapsulating node.
- 6to4 mechanism: A mechanism for IPv6 sites to communicate with each other over the IPv4 network without explicit tunnel setup, and for them to communicate with native IPv6 domains via relay routers. Effectively it treats the wide area IPv4 networks as a unicast point-to-point link layer.

3.5.1.1 Dual IP Layer Operation

An IPv6 node enabled with dual IP operation is provided with complete IPv4 implementation in addition to its own IPv6 layer functionalities. Such nodes are usually known as IPv6/IPv4 nodes since they have the dual ability to interoperate not only with the IPv6 nodes but also with the IPv4 nodes. While communicating with IPv4 nodes, IP4/IPv6 nodes use IPv4 packets whereas for IPv6 nodes, they use IPv6 packets.

3.5.1.2 Configured Tunneling Mechanism

Tunneling mechanism provides a way to utilize IPv4 router infrastructure to carry IPv6 traffic. An IPv4 tunnel refers to a chain of intervening IPv4 routers along the route of the IPv6 datagram. Both the starting and the ending nodes of a tunnel are essentially IPv6/IPv4 routers enabled with dual-stack implementation. The starting node of a tunnel encapsulates the IPv6 packet in IPv4 datagram that is decapsulated at the ending node of the tunnel. For each tunnel, the encapsulating node can be manually configured to store the attached IPv4 tunnel's end point address, which is associated with the tunnel-ending IPv4/IPv6 node's tunnel-connecting interface. While transmitting IPv6 datagram over the associated IPv4 tunnel, the encapsulating node encapsulates it in IPv4 datagram and route the packet to the terminating IPv4/IPv6 node using the configured tunnel end point address. The determination of which packets to tunnel is usually done via a routing table maintained by the encapsulating node. Tunnels with bi-directional trafficking ability behave as virtual point-to-point link. Any routing application for IPv6, which is running on an IPv4 tunnel end, will consider the other tunnel end point as one hop away and it will use the tunnel to send as well as receive control packets to get the routing information from other end.

Tunneling can be used in a variety of ways:

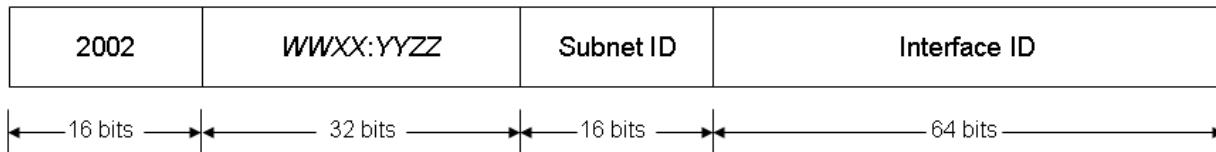
- Router-to-Router: IPv6/IPv4 routers interconnected by an IPv4 infrastructure can tunnel IPv6 packets between themselves. In this case, the tunnel spans one segment of the end-to-end path that the IPv6 packet takes.
- Host-to-Router: IPv6/IPv4 hosts can tunnel IPv6 packets to an intermediary IPv6/IPv4 router that is reachable via an IPv4 infrastructure. This type of tunnel spans the first segment of the packet's end-to-end path.
- Host-to-Host: IPv6/IPv4 hosts that are interconnected by an IPv4 infrastructure can tunnel IPv6 packets between themselves. In this case, the tunnel spans the entire end-to-end path of the packet.
- Router-to-Host: IPv6/IPv4 routers can tunnel IPv6 packets to their final destination IPv6/IPv4 host. This tunnel spans only the last segment of the end-to-end path.

3.5.1.3 6to4 Automatic Tunneling Mechanism

6to4 tunneling (RFC 3056 / Connection of IPv6 Domains via IPv4 Clouds) uses a simple mechanism to create automatic tunnels. Each node with a global unique IPv4 address is a 6to4 tunnel endpoint (if no IPv4 firewall prohibits traffic). 6to4 tunneling is mostly not a one-to-one tunnel. Also, a special IPv6 address indicates that this node will use 6to4 tunneling for connecting the world-wide IPv6 network.

Structure of a 6to4 Address

In order to use the 6to4 automatic tunneling mechanism, the network address must be specified in the following format:



where

WWXX: YYZZ Colon-hexadecimal representation of a public IPv4 address (w.x.y.z) assigned to a subnet or host.

3.5.2 Features and Assumptions

This section describes the implemented features, omitted features, assumptions and limitations of the Dual IP model.

3.5.2.1 Implemented Features

- Dual IP Layer Operation at Node Level
 - Support of both IPv4 and IPv6 stack on all the interfaces of a Dual-IP node
 - Support of all routing protocols at Dual-IP enabled node
 - Support for both wired and wireless network
 - IPv6 Address Configuration
- Configured Tunneling
 - Support of both IPv6 and IPv4 tunnels
 - Support of routing over the tunnels
 - Encapsulation
 - Tunnel as one-hop virtual-link

3.5.2.2 Omitted Features

- Tunnel MTU and Fragmentation
- Handling ICMPv4 Errors
- Security Considerations
- Scenario with tunnel to IPv6 space
- Fragmented Scenarios

3.5.2.3 Assumptions and Limitations

- A Dual-IP enabled node is assumed to have connection with both IPv6 and IPv4 network.
- In 6to4 multihome scenarios, all 6to4 routers should be connected to same v4 network.

- In case, a single interface of a Dual-IP enabled node is configured for more than one IPv4-tunnel, split-horizon for RIPng remains disabled on that interface. This is due to RIPng limitation to distinguish multiple neighbors connected to a single interface while sending/learning route from them.
- Optional Address in Tunnel Configuration file is mandatory in the following cases:
 - For running OLSRv2, EIGRP and IGRP routing protocols over the tunnel.
 - To enable static multicast over the tunnel.
 - User enables static routes over the tunnel and specify optional outgoing interface in static route file.
- IPv6 stack will be disabled on the 6to4 enable interface.

3.5.3 Command Line Configuration

To enable Dual IP, include the following parameter in the scenario configuration (.config) file.

```
[<Qualifier>] NETWORK-PROTOCOL DUAL-IP
```

The scope of this parameter declaration can be Global, Node, Subnet, or Interface. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

Configuration Requirements

In order to use the 6to4 automatic tunneling mechanism, one or more 6to4 subnets must be configured. To configure a 6to4 subnet, the network address used in the SUBNET or LINK statement must follow the format described in [Table 3.5.1.3](#).

Example

```
SUBNET N64-2002:c0a8:6301:0001: {1 thru 5}
LINK N64-2002:c0a8:6301:0002: {5, 6}
```

Note: Different values (other than zero) must be assigned to the SLA field (bits 48 to 63) of subnet prefix in the above SUBNET and LINK statements.

Dual IP Parameters

[Table 3-25](#) shows the Dual IP configuration parameters. See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 3-25. Dual IP Parameters

| Parameter | Value | Description |
|---|---|---|
| IP-FRAGMENT-HOLD-TIME <i>Optional</i> Scope: Global, Node | Time <i>Range:</i> • > 0S <i>Default:</i> 15S | Default duration for which IP fragments are held in the IP queue before they are released. |
| TUNNEL-CONFIG-FILE <i>Optional</i> Scope: Global, Node | Filename | Name of the tunnel configuration file. The tunnel configuration file contains a description of tunnels. Note: If all nodes support Dual IP, then the tunnel configuration file is not needed. The format of the tunnel configuration file is described in Section 3.5.3.1 . |

TABLE 3-25. Dual IP Parameters (Continued)

| Parameter | Value | Description |
|--|---|---|
| IPv6-ENABLE-6to4-TUNNELING <i>Optional</i> Scope: Interface | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Indicates whether 6to4 tunneling is enabled on the interface. Note: If 6to4 tunneling is enabled on an interface, the IPv6 stack will be disabled on that interface because 6to4-enabled interfaces are assumed to not have any direct connectivity with IPv6 networks. |
| NETWORK-LAYER-STATISTICS <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Indicates whether statistics are collected for network protocols. |

3.5.3.1 Format of the Tunnel Configuration File

The Tunnel Configuration file specifies IP tunnels.

Each line in the Tunnel Configuration file has the following format (all parameters must be entered on the same line):

```
<node-id> <tunnel-type> <tunnel-id> <tunnel-start-address>
[<optional-start-address>] <tunnel-end-address>
[<optional-address-for-tunnel-end>]
[INHERIT-FROM <inherit-interface-index>]
[BANDWIDTH <tunnel-bandwidth>] [PROPAGATION-DELAY <propagation-delay>]
```

where

| | |
|---------------|---|
| <Node ID> | Node ID of the node on which the tunnel is to be configured. |
| <tunnel-type> | Tunnel type. This can be v4-tunnel (indicating an IPv4 tunnel) or v6-tunnel (indicating an IPv6 tunnel). |
| <tunnel-id> | Tunnel ID is an integer to uniquely specify a tunnel for a node. Note: This parameter is deprecated and is only for backward compatibility. |

| | |
|--|---|
| <tunnel-start-address> | Start address of the tunnel |
| <optional-start-address> | Optional start address of the tunnel, enclosed in "[" and "]". This entry is optional unless: <ul style="list-style-type: none"> • Static multicast is enabled over the tunnel, or • Static routes are enabled over the tunnel and the optional outgoing interface is specified in static route file |
| <tunnel-end-address> | End address of the tunnel |
| <optional-address-for-tunnel-end> | Optional end address of the tunnel, enclosed in "[" and "]". This entry is optional unless: <ul style="list-style-type: none"> • Static multicast is enabled over the tunnel, or • Static routes are enabled over the tunnel and the optional outgoing interface is specified in static route file |
| INHERIT_FROM <inherit-interface-index> | Index of the interface from which routing and other protocols are inherited. This entry is optional. |
| BANDWIDTH <tunnel-bandwidth> | Maximum bandwidth of the tunnel. This entry is optional. |
| PROPAGATION-DELAY <propagation-delay> | End-to-end propagation delay of the tunnel. This entry is optional. |

Note: Tunnels should be bi-directional; hence they need to be configured at both ends.

Examples

1. The following is an example of the specification of an IPv4 tunnel in the Tunnel Configuration file:

```
2      v4-tunnel    192.168.1.1      192.168.2.2      INHERIT-FROM 1
5      v4-tunnel    192.168.2.2      192.168.1.1      INHERIT-FROM 2
```

2. The following is an example of the specification of an IPv6 tunnel in the Tunnel Configuration file:

```
2      v6-tunnel        2000::1        2001::2
5      v6-tunnel        2001::2        2000::1
```

3.5.4 GUI Configuration

This section describes how to configure Dual-IP in the GUI.

Configuring Dual-IP Parameters

To configure the Dual-IP parameters, perform the following steps:

1. Go to one of the following locations:
 - To set properties for a specific wireless subnet, go to **Wireless Subnet Properties Editor > Network Layer > General**.
 - To set properties for a specific wired subnet, go to **Wired Subnet Properties Editor > General**.

- To set properties for a specific point-to-point link, go to **Point-to-point Link Properties Editor > Point-to-point Link Properties > Network Protocol**.
- To set properties for a specific node, go to **Default Device Properties Editor > Node Configuration > Network Layer**.
- To set properties for a specific interface of a node, go to one of the following locations:
 - **Interface Properties Editor > Interfaces > Interface # > Network Layer**.
 - **Default Device Properties Editor > Interfaces > Interface # > Network Layer**.

In this section, we show how to configure the Dual-IP parameters for a specific node using the Default Device Properties Editor. Parameters can be set in the other properties editors in a similar way.

2. Set **Network Protocol** to *Dual-IP* and set the dependent parameters listed in [Table 3-26](#).

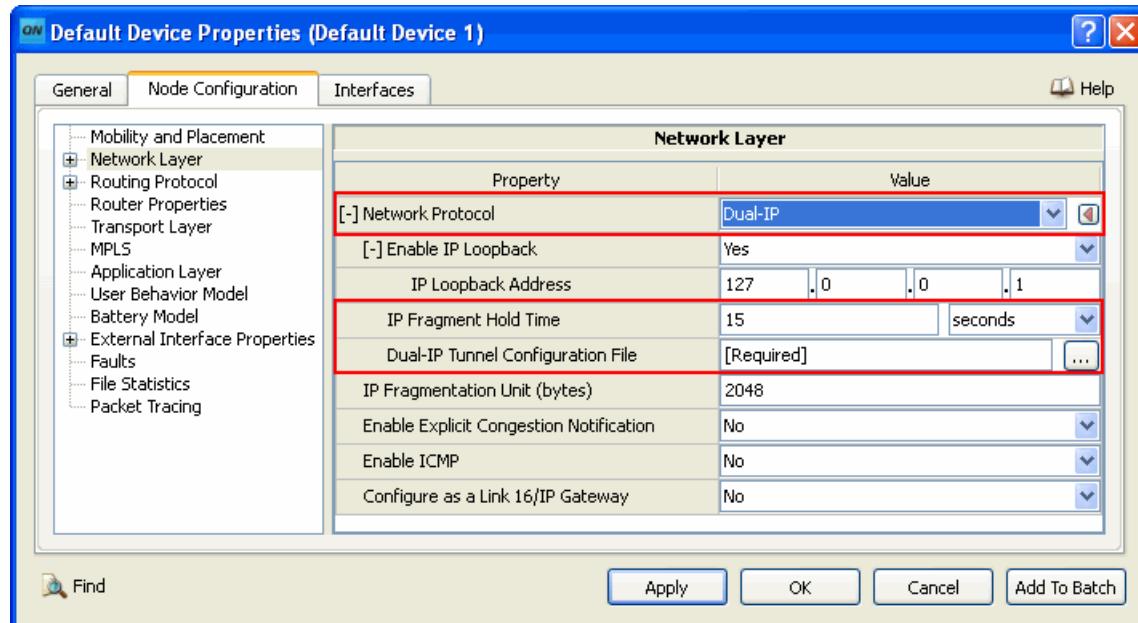


FIGURE 3-11. Setting Dual-IP Parameters

TABLE 3-26. Command Line Equivalent of Dual-IP Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-----------------------------------|------------------------|------------------------|
| IP Fragment Hold Time | Node | IP-FRAGMENT-HOLD-TIME |
| Dual-IP Tunnel Configuration File | Node | TUNNEL-CONFIG-FILE |

Setting Parameters

- Set **Dual-IP Tunnel Configuration File** to the name of the Tunnel Configuration file. The format of the Tunnel configuration file is described in [Section 3.5.3.1](#).

Enabling 6to4 Tunneling

The 6to4 Tunneling can be configured only at interface level.

To configure the 6to4 Tunneling in the GUI, Set **Network Protocol [=Dual-IP]** > **Enable IPv6 6to4 Tunneling** to Yes.

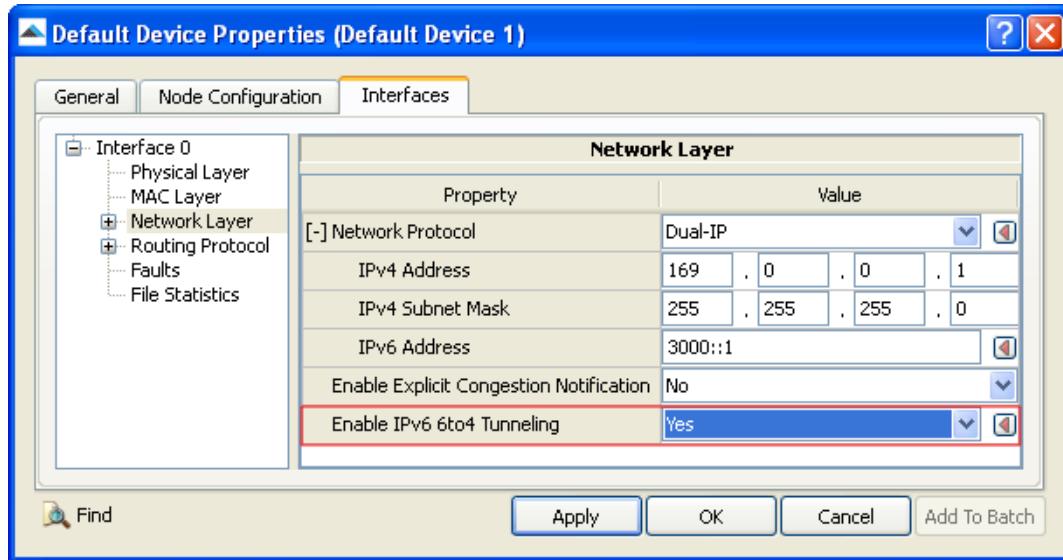


FIGURE 3-12. Enabling IPv6to4 Tunneling

TABLE 3-27. Command Line Equivalent of IPv6to4 Tunneling Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|--------------------------|------------------------|----------------------------|
| Enable IPv6to4 Tunneling | Interface | IPv6-ENABLE-6to4-TUNNELING |

Configuring Statistics Parameters

Statistics for Dual-IP can be collected at the global and node levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for Network Layer including Dual-IP, check the box labeled *Network* in the appropriate properties editor.

TABLE 3-28. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|--------------------------|
| Network | Global, Node | NETWORK-LAYER-STATISTICS |

3.5.5 Statistics

[Table 3-29](#) lists the Dual IP statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 3-29. Dual IP Statistics

| Statistic | Description |
|---|---|
| Broadcast Packets Sent at <type of tunnel> with Endpoint <tunnel endpoint> | Number of broadcast packets encapsulated at a tunneling interface with tunnel type and tunnel end-point. |
| Multicast Packets Sent at <type of tunnel> with Endpoint <tunnel endpoint> | Number of multicast packets encapsulated at a tunneling interface with tunnel type and tunnel end-point. |
| Unicast Packets Sent at <type of tunnel> with Endpoint <tunnel endpoint> | Number of unicast packets encapsulated at a tunneling interface with tunnel type and tunnel end-point. |
| Broadcast Packets Received at <type of tunnel> with Endpoint <tunnel endpoint> | Number of broadcast packets decapsulated at a tunneling interface with tunnel type and tunnel end-point. |
| Multicast Packets Received at <type of tunnel> with Endpoint <tunnel endpoint> | Number of multicast packets decapsulated at a tunneling interface with tunnel type and tunnel end-point. |
| Unicast Packets Received at <type of tunnel> with Endpoint <tunnel endpoint> | Number of unicast packets decapsulated at a tunneling interface with tunnel type and tunnel end-point. |
| Packets dropped at <type of tunnel> with Endpoint <tunnel endpoint> due to tunnel failure | Number of packets dropped due to tunnel failure at a tunneling interface with tunnel type and tunnel end-point. |

3.5.6 Scenarios Included in QualNet

The QualNet distribution includes several sample scenarios for the Dual IP model. All scenarios are located in the directory `QUALNET_HOME/scenarios/developer/dualip`. [Table 3-30](#) lists the sub-directory where each scenario is located.

TABLE 3-30. Dual IP Scenarios Included in QualNet

| Scenario | Description |
|--|---|
| dual-stack/dual-stack-v4-v6 | Shows the functionality of Dual-IP model for communication between a Dual-IP network with Pure IPv4 and Pure IPv6 Network. |
| dual-stack/wireless-dual-stack | Shows the functionality of Dual-IP model when dual-stack configured on a wireless subnet without any tunnel configuration. |
| multicast/all-dual-stack | Shows the Dual-IP implementation with multicast routing protocol in a Dual Stack network. |
| multicast/v4-multicast-v6-tunnel | Shows the functionality of Dual-IP model for Tunneling mechanism, configured with multicast routing protocol, using IPv6 as the Intermediate Network. |
| tunnel/6to4-automatic-tunnel/6to4-automatic-tunnel | Shows the functionality of Dual-IP model for 6to4 Automatic Tunneling mechanism between two IPv6 Networks interconnected by an intermediate IPv4 Network. |
| tunnel/6to4-automatic-tunnel/6to4-relay-router | Shows the functionality of Dual-IP model with 6to4 Automatic Tunneled Relay router as well as tunneled Dual-IP node. |
| tunnel/v4-tunnel/host-router-host | Shows the functionality of Dual-IP model for Tunneling mechanism, configured between a Dual-IP host and Dual-IP router, using IPv4 tunnel. |
| tunnel/v4-tunnel/multiple-v4-tunnels | Shows the functionality of Dual-IP model for Tunneling mechanism, configured between an IPv6 Network, using multiple IPv4 tunnel. |

TABLE 3-30. Dual IP Scenarios Included in QualNet (Continued)

| Scenario | Description |
|--------------------------------------|--|
| tunnel/v6-tunnel/multiple-v6-tunnels | Shows the functionality of Dual-IP model for Tunneling mechanism, configured between an IPv4 Network, using multiple IPv6 tunnel. |
| tunnel/v6-tunnel/router-v6-router | Shows the functionality of Dual-IP model for Tunneling mechanism, configured between IPv4/IPv6 (Dual-IP) routers, using IPv6 intermediate network. |

3.5.7 References

1. RFC 4213, "Basic Transition Mechanisms for IPv6 Hosts and Routers", E. Nordmark, R. Gilligan. October 2005.
2. RFC 2185, "Routing Aspects of IPv6 Transition", R. Callon, D. Haskin. September 1997.
3. RFC 3056, "Connection of IPv6 Domains via IPv4 Clouds", B. Carpenter, K. Moore. February 2001.
4. RFC 2471, "IPv6 Testing Address Allocation", R. Hinden, R. Fink, J. Postel. December 1998.
5. RFC 2462, "IPv6 Stateless Address Autoconfiguration", S. Thomson, T. Narten. December 1998.
6. RFC 2460, "Internet Protocol, Version 6 (IPv6) Specification", S. Deering, R. Hinden. December 1998.
7. RFC 2461, "Neighbor Discovery for IP Version 6 (IPv6)", T. Narten, E. Nordmark, W. Simpson. December 1998.
8. RFC 1191, "Path MTU Discovery", J. Mogul, S. Deering. November 1990.

3.6 Internet Protocol version 4 (IPv4)

The QualNet IPv4 model is based on RFC 791.

3.6.1 Description

IP is a protocol that provides a connectionless, best-effort delivery service of datagrams across the internet. Currently, there are two Internet Protocols: version 4 (IPv4) and version 6 (IPv6). For information on IPv6, see the section that immediately follows this section in this guide.

The Internet Protocol is designed for use in interconnected systems of packet-switched computer communication networks. The Internet Protocol provides for transmitting blocks of data called datagrams from sources to destinations, where sources and destinations are hosts identified by fixed length addresses. The Internet Protocol also provides for fragmentation and reassembly of long datagrams, if necessary, for transmission through *small packet* networks.

3.6.2 Command Line Configuration

To enable IPv4, specify the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] NETWORK-PROTOCOL IP
```

The scope of this parameter declaration can be Global, Node, Subnet, or Interface. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

IPv4 Parameters

The configuration parameters for IPv4 are shown in [Table 3-31](#). See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

Note: Refer to *QualNet User's Guide* for configuring additional Network layer parameters.

TABLE 3-31. IPv4 Parameters

| Parameter | Value | Description |
|--|---|--|
| IP-FRAGMENT-HOLD-TIME <i>Optional</i> Scope: Global, Node | Time <i>Range:</i> • > 0S <i>Default:</i> 15S | Default duration for which IP fragments are held in the IP queue before they are released. |
| NETWORK-LAYER-STATISTICS <i>Optional</i> Scope: Global, Node | List: • YES • NO <i>Default:</i> NO | Indicates whether statistics are collected for network protocols. |
| TRACE-IP <i>Optional</i> Scope: Global, Node | List: • YES • NO <i>Default:</i> NO | Indicates whether packet tracing is enabled for IPv4. Note: To enable packet tracing, some other parameters need to be configured as well. Refer to Section 4.2.10 of <i>QualNet User's Guide</i> for details. |

3.6.3 GUI Configuration

This section describes how to configure IPv4 in the GUI.

Note: Refer to *QualNet User's Guide* for configuring additional Network layer parameters.

Configuring IPv4 Parameters

To configure the IPv4 parameters, perform the following steps:

1. Go to one of the following locations:
 - To set wireless subnet properties, go to **Wireless Subnet Properties Editor** > **Network Layer** > **General**.
 - To set properties for a specific wired subnet, go to **Wired Subnet Properties Editor** > **General**.
 - To set properties for a specific point-to-point link, go to **Point-to-point link Properties Editor** > **Network Layer**.
 - To set properties for a specific node, go to **Node Properties Editor** > **Node Configuration** > **Network Layer**.
 - To set properties for a specific interface of a node, go to one of the following locations:
 - **Interface Properties Editor** > **Interfaces** > **Interface #** > **Network Layer**.
 - **Default Device Properties Editor** > **Interfaces** > **Interface #** > **Network Layer**.

In this section, we show how to configure IPv4 parameters for a specific node using the Default Device Properties Editor. Parameters can be set in the other properties editors in a similar way.

2. Set **Network Protocol** to *IPv4* and set the dependent parameters listed in Table 3-32.

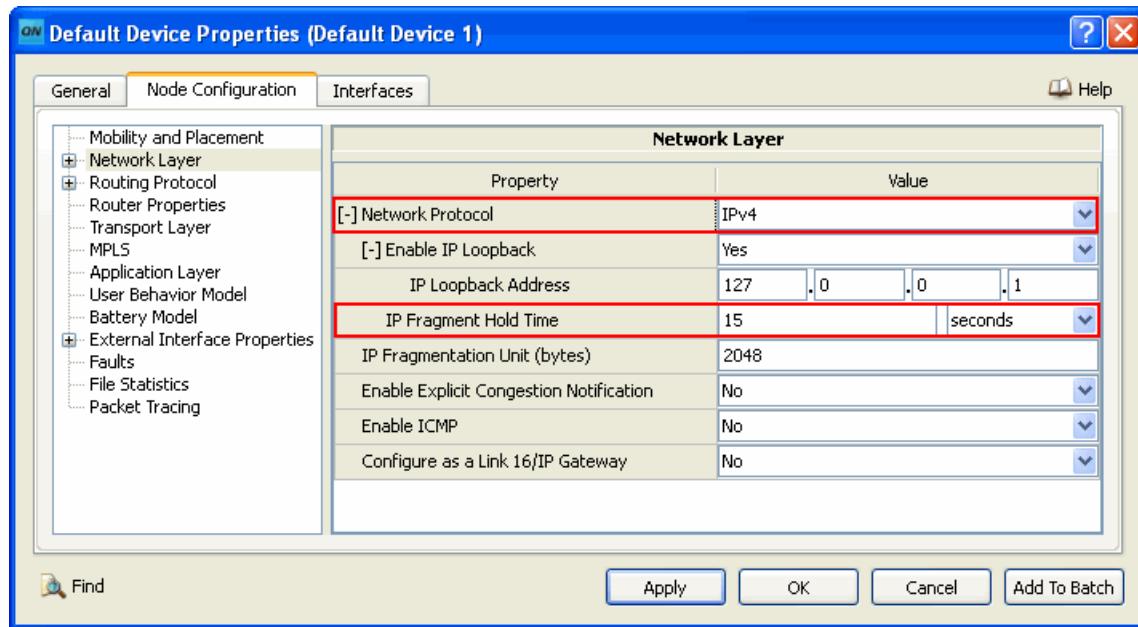


FIGURE 3-13. Setting IPv4 Protocol Parameters at Node Level

TABLE 3-32. Command Line Equivalent of IPv4 Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-----------------------|------------------------|------------------------|
| IP Fragment Hold Time | Node | IP-FRAGMENT-HOLD-TIME |

Configuring Statistics Parameters

Statistics for IPv4 can be collected at the global and node levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for Network Layer including IPv4, check the box labeled *Network* in the appropriate properties editor.

TABLE 3-33. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|--------------------------|
| Network | Global, Node | NETWORK-LAYER-STATISTICS |

Configuring Packet Tracing Parameters

Packet tracing for IPv4 can be enabled at the global and node levels. To enable packet tracing for IPv4, *Trace IP*. See Section 4.2.10 of *QualNet User's Guide* for details of configuring statistics parameters.

TABLE 3-34. Command Line Equivalent of Packet Tracing Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| Trace IPv4 | Global, Node | TRACE-IP |

3.6.4 Statistics

This section describes the file and dynamic statistics of the IPv4 model.

3.6.4.1 File Statistics

[Table 3-35](#) lists the IPv4 statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 3-35. IPv4 Statistics

| Statistic | Description |
|---|---|
| ipInHdrErrors | Packets dropped due to header errors. |
| ipInDelivers | Packets delivered to transport layer. |
| ipFragFails | Number of IP datagrams that have been discarded because they needed to be fragmented. |
| ipInUnknownProtos | Number of received IP datagrams discarded because of a lack of buffer space. |
| Packets fragmented | Number of packets fragmented. |
| Fragments created | Number of fragments created. |
| Fragments received | Number of fragments received. |
| Fragments dropped | Number of fragments dropped. |
| Fragments in Buffer | Number of fragments in buffer. |
| Fragments reassembled | Number of fragments reassembled. |
| Packets created after reassembling | Number of packets created after reassembling. |
| ipInDelivers TTL sum | TTL sum of the packets sent to upper layers. |
| ipRoutePktThruGt | Number of IP datagrams routed through gateway. |
| Send pkt to Other Net | Number of packets sent to other networks. |
| Recv pkt from Other Net | Number of packets received from other networks. |
| ipInDelivers TTL-based average hop count | TTL based on average hop count for packets sent to upper layers. |
| ipNumDroppedDueToBackplaneLimit | Number of packets dropped due to back plane limit. |
| Data packets sent unicast (packets) | Total number of unicast data packets sent. |
| Data packets received unicast (packets) | Total number of unicast data packets received. |
| Data packets forwarded unicast (packets) | Total number of unicast data packets forwarded. |
| Control packets sent unicast (packets) | Total number of unicast control packets sent. |
| Control packets received unicast(packets) | Total number of unicast control packets received. |
| Control packets forwarded unicast (packets) | Total number of unicast control packets forwarded. |
| Data bytes sent unicast (bytes) | Total number of unicast data bytes sent. |
| Data bytes received unicast (bytes) | Total number of unicast data bytes received. |
| Data bytes forwarded unicast (bytes) | Total number of unicast data bytes forwarded. |

TABLE 3-35. IPv4 Statistics (Continued)

| Statistic | Description |
|---|--|
| Control bytes sent unicast (bytes) | Total number of unicast control bytes sent. |
| Control bytes received unicast (bytes) | Total number of unicast control bytes received. |
| Control bytes forwarded unicast (bytes) | Total number of unicast control bytes forwarded. |
| Data packet sent broadcast (packets) | Total number of broadcast data packets sent. |
| Data packets received broadcast (packets) | Total number of broadcast data packets received. |
| Data packets forwarded broadcast (packets) | Total number of broadcast data packets forwarded. |
| Control packets sent broadcast (packets) | Total number of broadcast control packets sent. |
| Control packets received broadcast (packets) | Total number of broadcast control packets received. |
| Control packets forwarded broadcast (packets) | Total number of broadcast control packets forwarded. |
| Data bytes sent broadcast (bytes) | Total number of broadcast data bytes sent. |
| Data bytes received broadcast (bytes) | Total number of broadcast data bytes received. |
| Data bytes forwarded broadcast (bytes) | Total number of broadcast data bytes forwarded. |
| Control bytes sent broadcast (bytes) | Total number of broadcast control bytes sent. |
| Control bytes received broadcast (bytes) | Total number of broadcast control bytes received. |
| Control bytes forwarded broadcast (bytes) | Total number of broadcast control bytes forwarded. |
| Data packet sent multicast (packets) | Total number of multicast data packets sent. |
| Data packets received multicast (packets) | Total number of multicast data packets received. |
| Data packets forwarded multicast (packets) | Total number of multicast data packets forwarded. |
| Control packets sent multicast (packets) | Total number of multicast control packets sent. |
| Control packets received multicast (packets) | Total number of multicast control packets received. |
| Control packets forwarded multicast (packets) | Total number of multicast control packets forwarded. |
| Data bytes sent multicast (bytes) | Total number of multicast data bytes sent. |
| Data bytes received multicast (bytes) | Total number of multicast data bytes received. |
| Data bytes forwarded multicast (bytes) | Total number of multicast data bytes forwarded. |
| Control bytes sent multicast (bytes) | Total number of multicast control bytes sent. |
| Control bytes received multicast (bytes) | Total number of multicast control bytes received. |
| Control bytes forwarded multicast (bytes) | Total number of multicast control bytes forwarded. |
| Carried load (bits/second) | Carried load at the network layer. |
| Packets dropped due to no route (packets) | Total number of packets dropped due to no route. |
| Packets dropped due to expired TTL (packets) | Total number of packets dropped due to expired TTL. |
| Packets dropped due to queue overflow (packets) | Total number of packets dropped due to queue overflow. |
| Packets dropped due to other reasons (packets) | Total number of packets dropped due to other reasons. |
| Average delay (seconds) | Average delay at the network layer. |
| Average jitter (seconds) | Average jitter at the network layer. |
| Average hop count for data packets (hops) | Average hop count for data packets. |
| Average hop count for control packets (hops) | Average hop count for control packets. |
| Carried load unicast (bits/second) | Carried load for unicast packets. |
| Carried load multicast (bits/second) | Carried load for multicast packets. |
| Carried load broadcast (bits/second) | Carried load for broadcast packets. |
| Unicast Packets dropped due to no route (packets) | Total number of unicast packets dropped due to no route. |

TABLE 3-35. IPv4 Statistics (Continued)

| Statistic | Description |
|---|--|
| Multicast Packets dropped due to no route (packets) | Total number of multicast packets dropped due to no route. |
| Average delay unicast (seconds) | Average delay for unicast packets. |
| Average delay multicast (seconds) | Average delay for multicast packets. |
| Average delay broadcast (seconds) | Average delay for broadcast packets. |
| Average jitter unicast (seconds) | Average jitter for unicast packets. |
| Average jitter multicast (seconds) | Average jitter for multicast packets. |
| Average jitter broadcast (seconds) | Average jitter for broadcast packets. |
| Originated Carried load (bits/second) | Originated carried load at the network layer. |
| Forwarded Carried load (bits/second) | Forwarded carried load at the network layer. |

3.6.4.2 Database Statistics

In addition to the file statistics, the IPv4 model also enters statistics in various scenario statistics database tables. Refer to *QualNet Statistics Database User's Guide* for details.

3.6.4.3 Dynamic Statistics

The following dynamic statistics are enabled for the IPv4 model (refer to Chapter 5 of *QualNet User's Guide* for details of viewing dynamic statistics in the GUI during the simulation):

- Number of Packets Received
- Number of Packets Received with Errors in IP Header

3.6.5 References

1. RFC 791, "Internet Protocol Darpa Internet Program Protocol Specification", Information Science Institute, University of Southern California.

3.7 Internet Protocol version 6 (IPv6)

The QualNet IPv6 model is based on RFC 2460 and RFC 3513.

3.7.1 Description

IPv6 is the newer version of the Internet Protocol that is meant to replace IPv4. IPv4 has been used since the beginning of the Internet and has worked very well, but it has some serious limitations that IPv6 has been designed to overcome. Basic improvements of IPv6 are:

- Increased addressing space.
- IP address is expanded from 32 bits to 128 bits.
- Similar to IPv4, IPv6 also provides functionalities such as forwarding, fragmentation, automatic discovery of default routers, fastest forwarding process using destination cache, neighbor cache and prefix list, hierarchical networking, and simplified packet format.

IPv6 Addresses

IPv6 addresses are 128 bits long and are of three types. IP addresses are assigned to interfaces of the node. All interfaces must have at least one link-local unicast address. An interface may also have multiple IPv6 addresses of any type. There are three conventional forms to represent IPv6 address.

- The preferred form is `x:x:x:x:x:x:x:x`, where `x` is 16 bit hexadecimal value.
Example: `1080:0:0:0:8:800:200C:417A`
- Truncate a series of 0's with "`::`".
Example: The previous address can be represented as `1080::8:800:200C:417A`
- Mixed environment of IPv6 and IPv4 - `x:x:x:x:x:d.d.d.d` where '`x`' is the 16 bit hexadecimal value and '`d`' is the 8 bit decimal value as represented by IPv4.
Example: `0:0:0:0:0:13.1.68.3`

The following format is used to indicate the length of the subnet mask:

`<IPv6-address>/<prefix-length>`

where

`<IPv6-address>` IPv6 address in any of the above formats

`<prefix-length>` Number of bits in the subnet mask

IPv6 Unicast Address Types

A unicast address has two parts: subnet prefix and interface ID. Depending on the type of unicast address the length of these two types is different. There are several types of unicast address, such as:

- Global unicast address (allowed beyond the intranet)
- MSAP address (not used)
- IPX address (not used)
- site-local address (intranet specific address)
- link-local address (used for a single link)
- IPv4 capable address (`0:0:0:0:0:<IPv4 address>` or `0:0:0:0:FFFF:<IPv4 address>`)

- Unspecified address (0:0:0:0:0:0:0) indicates absence of an address
- Loopback address (0:0:0:0:0:0:1) used to send a message to itself

3.7.2 Features and Assumptions

This section describes the implemented features, omitted features, assumptions and limitations of the IPv6 model.

3.7.2.1 Implemented Features

- All types of IPv6 addressing
- NDP for Neighbor Discovery
- ICMPv6 for IPv6 control packets
- Fragmentation and Reassembly for large IPv6 packets
- Dual IP
- Static Multicasting

3.7.2.2 Omitted Features

- Support for MLD
- Router, Hop by Hop Option header, Destination Option header processing is blocked due to non-availability of other protocol or control blocks.
- Support for IPv6 Redirect Message

3.7.2.3 Assumptions and Limitations

- All neighbors receive the packet even if they are not the designated receiver. The receiver check if the packet was for itself, if not discards the packet.
- Link-Local and Site-Local addresses are automatically generated in QualNet so user should not specify this range of addresses to any Link or subnet.

3.7.3 Command Line Configuration

To enable IPv6, specify the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] NETWORK-PROTOCOL    IPv6
```

The scope of this parameter declaration can be Global, Node, Subnet, or Interface. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

IPv6 Parameters

The IPv6 configuration parameters are described in [Table 3-36](#). See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

Note: Refer to *QualNet User's Guide* for additional network layer parameters.

TABLE 3-36. IPv6 Parameters

| Parameter | Value | Description |
|--|---|--|
| IP-FRAGMENT-HOLD-TIME <i>Optional</i> Scope: Global, Node | Time <i>Range:</i> • > 0S <i>Default:</i> 15S | Default duration for which IP fragments are held in the IP queue before they are released. |
| NETWORK-LAYER-STATISTICS <i>Optional</i> Scope: Global, Node | List: • YES • NO <i>Default:</i> NO | Indicates whether statistics are collected for network protocols. |
| TRACE-IPV6 <i>Optional</i> Scope: Global, Node | List • YES • NO <i>Default:</i> NO | Indicates whether packet tracing is enabled for IPv6. Note: To enable packet tracing, some other parameters need to be configured as well. Refer to Section 4.2.10 of <i>QualNet User's Guide</i> for details. |

3.7.4 GUI Configuration

This section describes how to configure IPv6 in the GUI.

Configuring IPv6 Parameters

To configure the IPv6 parameters, perform the following steps:

1. Go to one of the following locations:
 - To set properties for a specific wireless subnet, go to **Wireless Subnet Properties Editor > Network Layer > General**.
 - To set properties for a specific wired subnet, go to **Wired Subnet Properties Editor > General**.
 - To set properties for a specific point-to-point link, go to **Point-to-point Link Properties Editor > Point-to-point Link Properties > Network Protocol**.
 - To set properties for a specific node, go to **Default Device Properties Editor > Node Configuration > Network Layer**.
 - To set properties for a specific interface of a node, go to one of the following locations:
 - **Interface Properties Editor > Interfaces > Interface # > Network Layer**
 - **Default Device Properties Editor > Interfaces > Interface # > Network Layer**.

In this section, we show how to configure IPv6 parameters for a specific node using the Default Device Properties Editor. Parameters can be set in the other properties editors in a similar way.

2. Set **Network Protocol** to *IPv6* and set the dependent parameters listed in [Table 3-37](#).

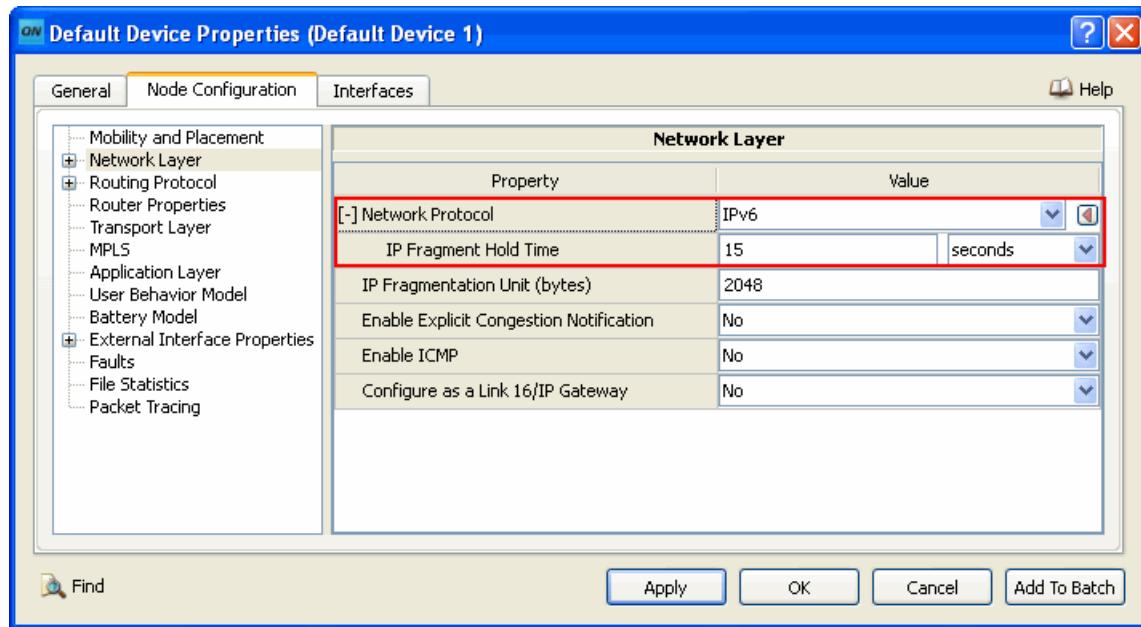


FIGURE 3-14. Configuring IPv6 Network Protocol Parameters

TABLE 3-37. Command Line Equivalent of IPv6 Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-----------------------|------------------------|------------------------|
| IP Fragment Hold Time | Node | IP-FRAGMENT-HOLD-TIME |

Note: Refer to *QualNet User's Guide* for configuring additional network layer parameters.

Configuring Statistics Parameters

Statistics for IPv6 can be collected at the global and node levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for Network Layer including IPv6, check the box labeled *Network* in the appropriate properties editor.

TABLE 3-38. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|--------------------------|
| Network | Global, Node | NETWORK-LAYER-STATISTICS |

Configuring Packet Tracing Parameters

Packet tracing for IPv6 can be enabled at the global and node levels. To enable packet tracing for IPv6, in addition to setting the IPv6 trace parameter, *Trace IPv6*, several other trace parameters also need to be set. See Section 4.2.10 of *QualNet User's Guide* for details of configuring packet tracing parameters.

TABLE 3-39. Command Line Equivalent of Packet Tracing Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| Trace IPv6 | Global, Node | TRACE-IPv6 |

3.7.5 Statistics

[Table 3-40](#) lists the IPv6 statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 3-40. IPv6 Statistics

| Statistic | Description |
|---|--|
| Total packets received | Number of packets received |
| Datagrams delivered to upper level | Number of datagrams delivered to upper level |
| Packets forwarded | Number of packets forwarded |
| Total ipv6 packets generated | Number of IPv6 packets generated |
| Packets rcvd for unreachable destination | Number of packets received for unreachable destination |
| Packets discarded due to no route | Number of packets discarded due to no route |
| Packets received from bad sources | Number of packets received from bad sources |
| Packets output discarded | Number of output IP packets discarded due to lack of buffer space |
| Number of packets received with improper version number | Number of packets received with incorrect version number in the IP header |
| Packets not forwarded because size greater than MTU | Number of packets not forwarded because size greater than MTU |
| Packet too short | Number of packets in which total packet length is less than the sum of header length and data length |
| Not enough data | Number of packets in which total packet length is less than the sum of all the header length |
| Number of fragmented packets received | Number of fragmented packets received |
| Number of fragmented packets dropped | Number of fragmented packets dropped |
| Number of fragmented packets time out | Number of fragmented packets time out |
| Number of Reassembled packets | Number of reassembled packets |
| Total output fragment created | Number of output fragment created |

3.7.6 Scenarios Included in QualNet

The QualNet distribution includes several sample scenarios for the IPv6 model. All scenarios are located in the directory `QUALNET_HOME/scenarios/developer/ipv6`. [Table 3-41](#) lists the sub-directory where each scenario is located.

TABLE 3-41. IPv6 Scenarios Included in QualNet

| Scenario | Description |
|-----------------------------------|---|
| addressing/link-addressing | Shows the functionality of IPv6 Addressing. |
| addressing/subnet-addressing | Shows the functionality of IPv6 Addressing. |
| fragmentation | Shows the functionality of fragmentation using IPv6. |
| ipv6_csma | Shows the functionality of WLAN for multiple subnets. |
| ipv6_pc-pc | Shows the functionality of WLAN for multiple subnets |
| mac-dot11 | Shows the functionality of mac-dot11 for IPv6. |
| mac-dot11e | Shows the functionality of mac-dot11e for IPv6. |
| ndp-functionality/ndp-link | Shows the NDP functionality of IPv6. |
| ndp-functionality/ndp-subnet | Shows the NDP functionality of IPv6. |
| ndp-functionality/ndp-subnet-link | Shows the NDP functionality of IPv6. |
| static-multicast | Shows the functionality of static multicast using IPv6. |
| static-routing | Shows the functionality static routing using IPv6. |
| switch-ipv6 | Shows the switch configuration of IPv6. |

3.7.7 References

1. RFC 2460, “Internet Protocol, Version 6 (IPv6) Specification”, S. Deering, R. Hinden. December 1998.
2. RFC 3513, “Internet Protocol Version 6 (IPv6) Addressing Architecture”, R. Hinden, S. Deering. April 2003.

3.8 Neighbor Discovery Protocol

The QualNet Neighbor Discovery Protocol is based on the RFC 2461, RFC 2460, RFC 2462 and RFC 2463.

3.8.1 Description

The IPv6 Neighbor Discovery protocol corresponds to a combination of the IPv4 protocols ARP, ICMP Router Discovery (RDISC), and ICMP Redirect (ICMPv4).

Nodes (hosts and routers) use Neighbor Discovery to determine the link-layer addresses for neighbors known to reside on attached links and to quickly purge cached values that become invalid. Hosts also use Neighbor Discovery to find neighboring routers that are willing to forward packets on their behalf. Finally, nodes use the protocol to actively keep track of which neighbors are reachable and which are not, and to detect changed link-layer addresses.

Neighbor discovery defines mechanisms for solving each of the following problems:

- Router Discovery: How hosts locate routers that reside on an attached link.
- Prefix Discovery: How hosts discover the set of address prefixes that define which destinations are on-link for an attached link. (Nodes use prefixes to distinguish destinations that reside on-link from those only reachable through a router.)
- Parameter Discovery: How a node learns such link parameters as the link MTU or such Internet parameters as the hop limit value to place in outgoing packets.
- Address Auto configuration: How nodes automatically configure an address for an interface.
- Address Resolution: How nodes determine the link-layer address of an on-link destination (e.g., a neighbor) given only the destination's IP address.
- Next-hop Determination: The algorithm for mapping an IP destination address into the IP address of the neighbor to which traffic for the destination should be sent. The next-hop can be a router or the destination itself.
- Neighbor Unreachability Detection: How nodes determine that a neighbor is no longer reachable. For neighbors used as routers, alternate default routers can be tried. For both routers and hosts, address resolution can be performed again.
- Duplicate Address Detection: How a node determines that an address it wishes to use is not already in use by another node.
- Redirect: How a router informs a host of a better first-hop node to reach a particular destination.

Neighbor Discovery defines five different ICMP packet types: A pair of Router Solicitation and Router Advertisement messages, a pair of Neighbor Solicitation and Neighbor Advertisements messages, and a Redirect message. The messages serve the following purpose:

- Router Solicitation: When an interface becomes enabled, hosts may send out Router Solicitations that request routers to generate Router Advertisements immediately rather than at their next scheduled time.
- Router Advertisement: Routers advertise their presence together with various link and Internet parameters either periodically, or in response to a Router Solicitation message. Router Advertisements contain prefixes that are used for on-link determination and/or address configuration, a suggested hop limit value, etc.
- Neighbor Solicitation: Sent by a node to determine the link-layer address of a neighbor, or to verify that a neighbor is still reachable via a cached link-layer address. Neighbor Solicitations are also used for Duplicate Address Detection.

- Neighbor Advertisement: A response to a Neighbor Solicitation message. A node may also send unsolicited Neighbor Advertisements to announce a link-layer address change.
- Redirect: Used by routers to inform hosts of a better first hop for a destination which has been discussed in the ICMPv6 section as well.

3.8.2 Features and Assumptions

This section describes the implemented features, omitted features, assumptions and limitations of the Neighbor Discovery protocol.

3.8.2.1 Implemented Features

- Router Solicitation
- Router Advertisements
- Neighbor Solicitation
- Neighbor Advertisements
- Redirect message
- Option Features (Link-layer address, Prefix information, Redirected header, MTU).
- Neighbor Unreachability Detection for garbage collection of stale and unused entries.

3.8.2.2 Omitted Features

- Proxy Neighbor Advertisements.
- Anycast Neighbor Advertisements.
- Manual configuration of different variables for host and routers.
- Address Autoconfiguration mechanism.
- Duplicate Address Detection mechanism.
- Handling link-layer address change.
- Sending Unsolicited Neighbor Advertisements.
- Policy for Selecting Router from Default Router List.
- Inbound load balancing.

3.8.2.3 Assumptions and Limitations

- The M and O bits under ICMP fields in Router Advertisement message are not set.
- Router lifetime is not considered.
- The L and A bits under Prefix Information fields in Router Advertisement message are not set. 4) Valid lifetime and preferred lifetime are not considered.
- Link-Layer Address is created using 24 bits of node id and 8 bits of interface id.
- Security associations are assumed to be not present.

3.8.3 Command Line Configuration

The Neighbor Discovery protocol is automatically enabled if IPv6 is enabled (see [Section 3.7](#). for details).

Neighbor Discovery Protocol Parameters

[Table 3-42](#) shows the Neighbor Discovery Protocol configuration parameters. See [Section 1.2.1.3](#) for a description of the format used for the parameter table

TABLE 3-42. Neighbor Discovery Protocol Parameters

| Parameter | Value | Description |
|--|--|--|
| NDP-STATISTICS <i>Optional</i> Scope: Global, Node | List <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Indicates whether statistics are collected for NDP protocol. |

3.8.4 GUI Configuration

The Neighbor Discovery protocol is automatically enabled if IPv6 is enabled (see [Section 3.7](#). for details).

Configuring Statistics Parameters

Statistics for the Neighbor Discovery protocol can be collected at the global, node, and interface levels. See [Section 4.2.9 of QualNet User's Guide](#) for details of configuring statistics parameters.

To enable statistics collection for the Neighbor Discovery protocol, check the box labeled **NDP** in the appropriate properties editor.

TABLE 3-43. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| NDP | Global, Node | NDP-STATISTICS |

3.8.5 Statistics

[Table 3-44](#) lists the Neighbor Discovery protocol statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 3-44. Neighbor Discovery Protocol Statistics

| Statistic | Description |
|---|---|
| Number of Router Solicitation Messages Sent | Number of Router Solicitation Messages Sent |
| Number of Router Advertisement Messages Sent | Number of Router Advertisement Messages Sent |
| Number of Neighbor Solicitation Messages Sent | Number of Neighbor Solicitation Messages Sent |
| Number of Neighbor Advertisement Messages Sent | Number of Neighbor Advertisement Messages Sent |
| Number of Redirect Messages Sent | Number of Redirect Messages Sent |
| Number of Router Solicitation Messages Received | Number of Router Solicitation Messages Received |

TABLE 3-44. Neighbor Discovery Protocol Statistics (Continued)

| Statistic | Description |
|--|--|
| Number of Router Advertisement Messages Received | Number of Router Advertisement Messages Received |
| Neighbor Solicitation Messages Received | Number of Neighbor Solicitation Messages Received |
| Neighbor Advertisement Messages Received | Number of Neighbor Advertisement Messages Received |
| Number of Invalid Router Solicitation Messages Received | Number of Bad Router Solicitation Messages Received |
| Number of Invalid Router Advertisement Messages Received | Number of Bad Router Advertisement Messages Received |
| Number of Invalid Neighbor Solicitation Messages Received | Number of Bad Neighbor Solicitation Messages Received |
| Number of Invalid Neighbor Advertisement Messages Received | Number of Bad Neighbor Advertisement Messages Received |
| Number of Invalid Redirect Messages Received | Number of Bad Redirect Messages Received |

3.8.6 Scenarios Included in QualNet

The QualNet distribution includes several sample scenarios for the Neighbor Discovery protocol. All scenarios are located in the directory `QUALNET_HOME/scenarios/developer/ipv6/ndp-functionality`. [Table 3-45](#) lists the sub-directory where each scenario is located.

TABLE 3-45. Neighbor Discovery Protocol Scenarios Included in QualNet

| Scenario | Description |
|-----------------|--|
| ndp-link | Shows the NDP functionality of IPv6 when all the nodes are connected through point-to-point links. |
| ndp-subnet | Shows the NDP functionality of IPv6 when two different subnets are connected through two routers. |
| ndp-subnet-link | Shows the NDP functionality of IPv6 when a single subnet is connected with two routers to two different links. |

3.8.7 References

1. RFC 2461, “Neighbor Discovery for IP Version 6 (IPv6)”, T. Narten, E. Nordmark, W. Simpson. December 1998
2. RFC 2460, “Internet Protocol, Version 6 (IPv6) Specification”, S. Deering, R. Hinden. December 1998.
3. RFC 2462, “IPv6 Stateless Address Autoconfiguration”, S. Thomson, T. Narten. December 1998.
4. RFC 2463, “Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification”, A. Conta, S. Deering. December 1998.

4

Unicast Routing Protocol Models

This chapter describes features, configuration requirements and parameters, statistics, and scenarios for Unicast Routing Protocol Models, and consists of the following sections:

- Bellman-Ford Routing Protocol
- Routing Information Protocol next generation (RIPng)
- Routing Information Protocol/Routing Information Protocol version 2 (RIP/RIPv2)
- Static and Default Routes

4.1 Bellman-Ford Routing Protocol

4.1.1 Description

This is a generic Bellman-Ford (a.k.a. Ford Fulkerson) routing algorithm. This algorithm is the underlying mechanism of Routing Information Protocol (RIP), but this protocol is not RIPv2-compliant. It is a distance vector routing algorithm that uses the User Datagram Protocol (UDP) for control packet transmission.

4.1.2 Command Line Configuration

To select Bellman-Ford as the routing protocol, include the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] ROUTING-PROTOCOL BELLMANFORD
```

The scope of this parameter declaration can be Global, Node, Subnet, or Interface. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

Configuration Requirements

In order to use Bellman-Ford, IPv4 must be enabled, i.e., NETWORK-PROTOCOL must be set to IPv4, DUAL-IP, CELLULAR-LAYER3, or GSM-LAYER3.

Bellman-Ford Parameters

[Table 4-1](#) shows the Bellman-Ford configuration parameters. See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 4-1. Bellman-Ford Parameters

| Parameter | Value | Description |
|--|--|--|
| ROUTING-STATISTICS <i>Optional</i> Scope: Global, Node | List: • YES • NO <i>Default: NO</i> | Indicates whether statistics are collected for routing protocols. |
| TRACE-BELLMANFORD <i>Optional</i> Scope: Global, Node | List: • YES • NO <i>Default: NO</i> | Indicates whether packet tracing is enabled for Bellman-Ford. Note: To enable packet tracing, some other parameters need to be configured as well. Refer to Section 4.2.10 of <i>QualNet User's Guide</i> for details. |

4.1.3 GUI Configuration

This section describes how to configure Bellman Ford in the GUI.

Configuring Bellman Ford Parameters

To configure the Bellman Ford parameters, perform the following steps:

1. Go to one of the following locations:
 - To set properties for a specific wireless subnet, go to **Wireless Subnet Properties Editor > Routing Protocol > General**.
 - To set properties for a specific wired subnet, go to **Wired Subnet Properties Editor > Routing Protocol > General**.
 - To set properties for a specific point-to-point link, go to **Point-to-point Link Properties Editor > Point-to-point Link Properties > Routing Protocol**.
 - To set properties for a specific node, go to **Default Device Properties Editor > Node Configuration > Routing Protocol**.
 - To set properties for a specific interface of a node, go to one of the following locations:
 - **Interface Properties Editor > Interfaces > Interface # > Routing Protocol**
 - **Default Device Properties Editor > Interfaces > Interface # > Routing Protocol**.

In this section, we show how to configure Bellman Ford parameters for a specific node using the Default Device Properties Editor. Parameters can be set in the other properties editors in a similar way.

2. Set **Routing Protocol IPv4** to *Bellman Ford*.

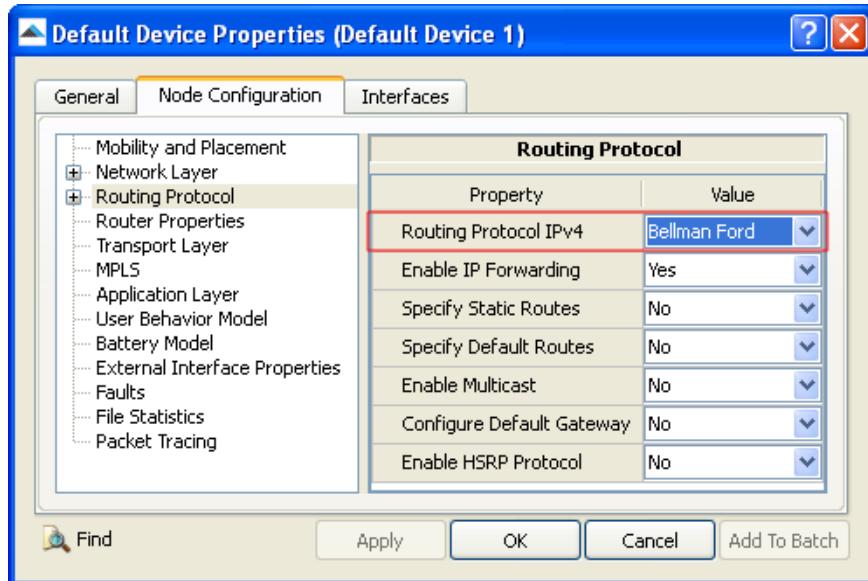


FIGURE 4-1. Setting Routing Protocol to Bellman Ford

Configuring Statistics Parameters

Statistics for Bellman Ford can be collected at the global and node levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for routing protocols including Bellman Ford, check the box labeled **Routing** in the appropriate properties editor.

TABLE 4-2. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| Routing | Global, Node | ROUTING-STATISTICS |

Configuring Packet Tracing Parameters

Packet tracing for Bellman Ford can be enabled at the global level. To enable packet tracing for Bellman Ford, in addition to setting the Bellman Ford trace parameter, **Trace BELLMANFORD**, several other trace parameters also need to be set. See Section 4.2.10 of *QualNet User's Guide* for details of configuring packet tracing parameters.

TABLE 4-3. Command Line Equivalent of Packet Tracing Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-------------------|------------------------|------------------------|
| Trace BELLMANFORD | Global, Node | TRACE-BELLMANFORD |

4.1.4 Statistics

[Table 4-4](#) lists the Bellman-Ford statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 4-4. Bellman-Ford Statistics

| Statistic | Description |
|-----------------------------------|---|
| Number of periodic updates sent | Total number of periodic update messages sent. |
| Number of triggered updates sent | Total number of triggered update messages sent. |
| Number of route timeouts | Total number of route timeouts occurred. |
| Number of update packets received | Total number of update packets received. |

4.2 Routing Information Protocol next generation (RIPng)

4.2.1 Description

RIPng is a proactive *Interior Gateway Protocol* based on the distance-vector algorithm. RIPng is intended for use within the IPv6-based Internet.

As it is a distance-vector routing protocol, it forms routing tables by exchanging routing table information with each router. There are two types of updates. One is a Regular update, which is periodically sent and contains the whole routing table information. The other is a Triggered update, which is sent when a router's routing table changes and contains only those routing entities which have been modified. When a router receives a packet, it updates its routing table and if its routing table has changed, it sends a triggered update to its neighbor router.

4.2.2 Command Line Configuration

To select RIPng as the routing protocol, specify the following parameter(s) in the scenario configuration (.config) file:

- For a dual IP-node, use the following parameter:

```
[<Qualifier>] ROUTING-PROTOCOL-IPv6      RIPng
```

- For an IPv6 node, use *either* of the following parameters:

```
[<Qualifier>] ROUTING-PROTOCOL      RIPng
```

or

```
[<Qualifier>] ROUTING-PROTOCOL-IPv6      RIPng
```

The scope of this parameter declaration can be Global, Node, Subnet, or Interface. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

Configuration Requirements

In order to use RIPng, IPv6 must be enabled, i.e., NETWORK-PROTOCOL must be set to IPv6 or DUAL-IP.

RIPng Parameters

Table 4-5 shows the parameters used by RIPng. See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 4-5. RIPng Parameters

| Parameters | Value | Description |
|--|--|--|
| SPLIT-HORIZON <i>Optional</i> Scope: All | List: <ul style="list-style-type: none">• NO• SIMPLE• POISONED-REVERSE <i>Default:</i> SIMPLE | Split horizon method used to prevent route loops. NO : No split horizon SIMPLE : When sending route updates to a neighbor, omit those entries that are learned from that neighbor (i.e., routes with that neighbor as the next hop) POISONED-REVERSE : Instead of omitting those routes as in the simple split horizon scheme, those route entries will be included in the route updates. However, the metrics of those routes are set to infinity. |
| ROUTING-STATISTICS <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Indicates whether routing protocol statistics are collected. |
| TRACE-RIPNG <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Indicates whether packet tracing is enabled for RIPng. Note: To enable packet tracing, some other parameters need to be configured as well. Refer to Section 4.2.10 of <i>QualNet User's Guide</i> for details. |

4.2.3 GUI Configuration

This section describes how to configure RIPng in the GUI.

Configuring RIPng Parameters

To configure the RIPng parameters, perform the following steps:

1. Go to one of the following locations:
 - To set properties for a specific wireless subnet, go to **Wireless Subnet Properties Editor > Routing Protocol > General**.
 - To set properties for a specific wired subnet, go to **Wired Subnet Properties Editor > Routing Protocol > General**.
 - To set properties for a specific point-to-point link, go to **Point-to-point Link Properties Editor > Point-to-point Link Properties > Routing Protocol**.

- To set properties for a specific node, go to **Default Device Properties Editor > Node Configuration > Routing Protocol**.
- To set properties for a specific interface of a node, go to one of the following locations:
 - **Interface Properties Editor > Interfaces > Interface # > Routing Protocol**
 - **Default Device Properties Editor > Interfaces > Interface # > Routing Protocol**.

In this section, we show how to configure RIPng parameters for a specific node using the Default Device Properties Editor. Parameters can be set in the other properties editors in a similar way.

2. Set **Routing Protocol IPv6** to *RIPng* and set the dependent parameters listed in [Table 4-6](#).

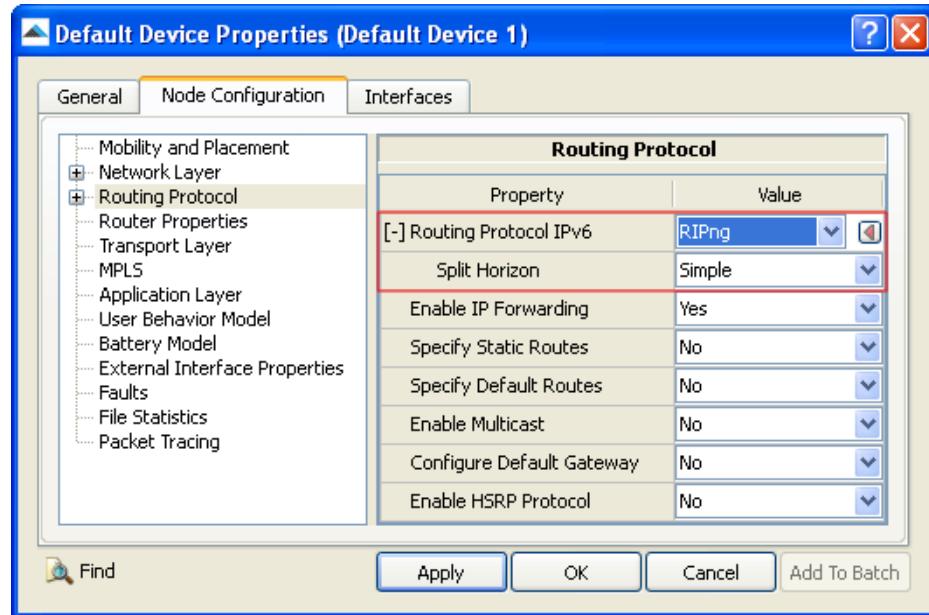


FIGURE 4-2. Setting Routing Protocol to RIPng

TABLE 4-6. Command Line Equivalent of RIPng Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|-------------------------|------------------------|
| Split Horizon | Node, Subnet, Interface | SPLIT-HORIZON |

Configuring Statistics Parameters

Statistics for RIPng can be collected at the global and node levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for routing protocols including RIPng, check the box labeled **Routing** in the appropriate properties editor.

TABLE 4-7. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|----------------------|-------------------------------|-------------------------------|
| Routing | Global, Node | ROUTING-STATISTICS |

Configuring Packet Tracing Parameters

Packet tracing for RIPng can be enabled at the global level. To enable packet tracing for RIPng, in addition to setting the RIPng trace parameter, **Trace RIPng**, several other trace parameters also need to be set. See Section 4.2.10 of *QualNet User's Guide* for details of configuring packet tracing parameters.

TABLE 4-8. Command Line Equivalent of Packet Tracing Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|----------------------|-------------------------------|-------------------------------|
| TRACE RIPng | Global | TRACE-RIPNG |

4.2.4 Statistics

This section describes the file and dynamic statistics of the RIPng model.

4.2.4.1 File Statistics

[Table 4-9](#) lists the RIPng statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 4-9. RIPng Statistics

| Statistic | Description |
|-----------------------|--|
| RegularUpdateEvents | Number of regular update events. |
| TriggeredUpdateEvents | Number of triggered update events. |
| RouteTimeouts | Number of times route time out occurred. |
| ResponsesReceived | Number of response packets received. |
| RegularPacketSent | Number of regular updates packet sent. |
| TriggeredPacketSent | Number of triggered updates packet sent. |

4.2.4.2 Dynamic Statistics

The following dynamic statistics are enabled for the RIPng model (refer to Chapter 5 of *QualNet User's Guide* for details of viewing dynamic statistics in the GUI during the simulation):

- Number of Regular Update Events
- Number of Triggered Update Events
- Number of Route Timeout Events
- Number of Received Response Packets
- Number of Regular Update Packets Sent

- Number of Triggered Update Packets Sent

4.2.5 Scenarios Included in QualNet

The QualNet distribution includes several sample scenarios for the RIPng protocol. All scenarios are located in the directory `QUALNET_HOME/scenarios/developer/ripng`. [Table 4-10](#) lists the sub-directory where each scenario is located.

TABLE 4-10. RIPng Scenarios Included in QualNet

| Scenario | Description |
|-----------|--|
| sample-1 | Shows RIPng behavior with respect to processing of response messages. |
| sample-2 | Shows the processing used to create response messages that contain all or part of the routing table. |
| sample-3 | Shows the RIPng behavior with respect to split horizon mechanism. |
| sample-4 | Shows the functionality of RIPng routing using IPv6. |
| sample-5 | Shows RIPng behavior with simple split horizon during single interface failure. |
| sample-6 | Shows RIPng behavior during interface failure. |
| sample-7 | Shows the RIPng behavior with split horizon mechanism with poisoned reverse. |
| sample-8 | Shows RIPng behavior during link failure. |
| sample-9 | Shows RIPng behavior for large number of hosts in a subnet. |
| sample-10 | Shows RIPng behavior without split horizon mechanism. |

4.3 Routing Information Protocol/Routing Information Protocol version 2 (RIP/RIPv2)

4.3.1 Description

RIP or RIPv2 are internet standard implementations of the Bellman-Ford (a.k.a. Ford Fulkerson) routing algorithm. It is a distance vector routing algorithm using the User Datagram Protocol (UDP) protocol for control packet transmission.

4.3.2 Command Line Configuration

To select RIP/RIPv2 as the routing protocol, include the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] ROUTING-PROTOCOL      RIP
```

The scope of this parameter declaration can be Global, Node, Subnet, or Interface. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

Configuration Requirements

In order to use RIP/RIPv2, IPv4 must be enabled, i.e., NETWORK-PROTOCOL must be set to IPv4, DUAL-IP, CELLULAR-LAYER3, or GSM-LAYER3.

RIP Parameters

[Table 4-11](#) describes the RIP/RIPv2 configuration parameters. See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 4-11. RIP/RIPv2- Parameters

| Parameter | Value | Description |
|--|---|--|
| RIP-VERSION <i>Optional</i> Scope: All | List: <ul style="list-style-type: none">• 1• 2 <i>Default:</i> 2 | Appropriate version of RIP is selected based on the value. |
| SPLIT-HORIZON <i>Optional</i> Scope: All | List: <ul style="list-style-type: none">• NO• SIMPLE• POISONED-REVERSE <i>Default:</i> SIMPLE | Split horizon method used to prevent route loops. NO : No split horizon SIMPLE : When sending route updates to a neighbor, omit those entries that are learned from that neighbor (i.e., routes with that neighbor as the next hop) POISONED-REVERSE : Instead of omitting those routes as in the simple split horizon scheme, those route entries will be included in the route updates. However, the metrics of those routes are set to infinity. |
| ROUTING-STATISTICS <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Indicates whether routing protocol statistics are collected. |

4.3.3 GUI Configuration

This section describes how to configure RIP/RIPv2 in the GUI.

Configuring RIP/RIPv2 Parameters

To configure the RIP/RIPv2 parameters, perform the following steps:

1. Go to one of the following locations:
 - To set properties for a specific wireless subnet, go to **Wireless Subnet Properties Editor > Routing Protocol > General**.
 - To set properties for a specific wired subnet, go to **Wired Subnet Properties Editor > Routing Protocol > General**.
 - To set properties for a specific point-to-point link, go to **Point-to-point Link Properties Editor > Point-to-point Link Properties > Routing Protocol**.
 - To set properties for a specific node, go to **Default Device Properties Editor > Node Configuration > Routing Protocol**.
 - To set properties for a specific interface of a node, go to one of the following locations:
 - **Interface Properties Editor > Interfaces > Interface # > Routing Protocol**
 - **Default Device Properties Editor > Interfaces > Interface # > Routing Protocol**.

In this section, we show how to configure RIP/RIPv2 parameters for a specific node using the Default Device Properties Editor. Parameters can be set in the other properties editors in a similar way.

- Set **Routing Protocol IPv4** to *RIP* and set the dependent parameters listed in [Table 4-12](#).

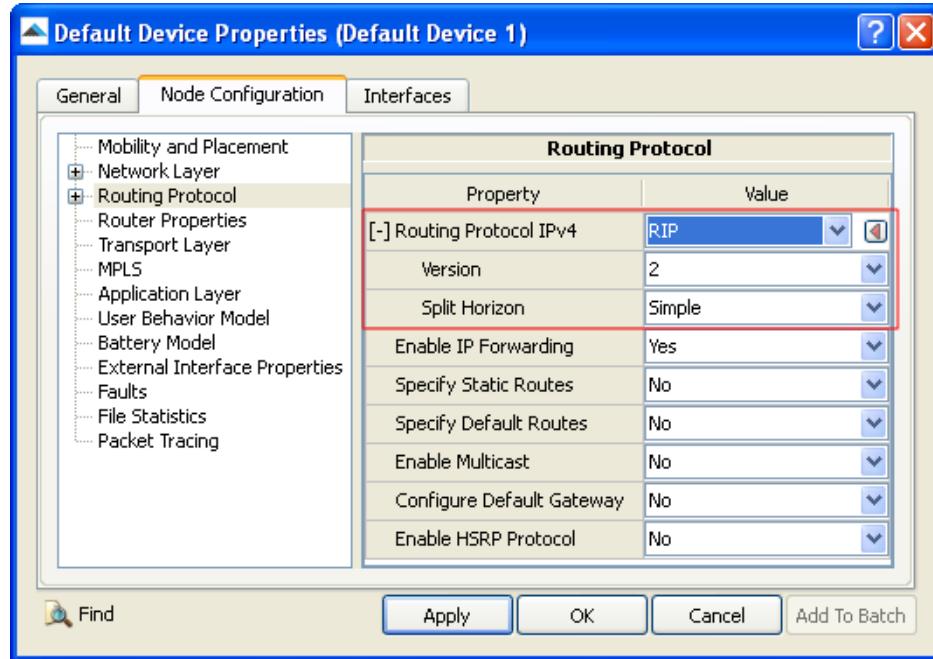


FIGURE 4-3. Setting Routing Protocol to RIP/RIPv2

TABLE 4-12. Command Line Equivalent of RIP/RIPv2 Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|-------------------------|------------------------|
| Version | Node, Subnet, Interface | RIP-VERSION |
| Split Horizon | Node, Subnet, Interface | SPLIT-HORIZON |

Configuring Statistics Parameters

Statistics for RIP/RIPv2 can be collected at the global and node levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for routing protocols including RIP/RIPv2, check the box labeled **Routing** in the appropriate properties editor.

TABLE 4-13. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| Routing | Global, Node | ROUTING-STATISTICS |

4.3.4 Statistics

This section describes the file and dynamic statistics of the RIP/RIPv2 model.

4.3.4.1 File Statistics

[Table 4-14](#) lists the RIP/RIPv2 statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 4-14. RIP/RIPv2 Statistics

| Statistic | Description |
|-------------------------------|---|
| Regular Update Events | Number of regular update events. |
| Triggered Update Events | Number of Triggered Update Events |
| Route Timeout Events | Number of Route Timeout Events |
| Response Packets Received | Number of Response Packets Received |
| Regular Update Packets Sent | Number of Regular Update Packets Sent |
| Triggered Update Packets Sent | Number of Triggered Update Packets Sent |

4.3.4.2 Dynamic Statistics

The following dynamic statistics are enabled for the RIP/RIPv2 model (refer to Chapter 5 of *QualNet User's Guide* for details of viewing dynamic statistics in the GUI during the simulation):

- Number of Regular Update Events
- Number of Triggered Update Events
- Number of Route Timeout Events
- Number of Received Response Packets
- Number of Regular Update Packets Sent
- Number of Triggered Update Packets Sent

4.3.5 Scenarios Included in QualNet

The QualNet distribution includes several sample scenarios for the RIP/RIPv2 model. All scenarios are located in the directory `QUALNET_HOME/scenarios/developer/rip` [Table 4-15](#) lists the sub-directory where each scenario is located.

TABLE 4-15. RIP/RIPv2 Scenarios Included in QualNet

| Scenario | Description |
|-----------------------------------|--|
| version-1/output-processing/bs1 | Shows the processing used to create response messages that contain all or part of the routing table. |
| version-1/output-processing/ns1 | Shows the processing used to create response messages that contain all or part of the routing table. |
| version-1/response-processing/ns1 | Shows the processing of response messages. |
| version-1/response-processing/ns3 | Shows the processing of response messages. |
| version-1/split-horizon/ns1 | Shows the split horizon mechanism. |
| version-1/split-horizon/ns2 | Shows the split horizon mechanism. |
| version-2/output-processing/bs1 | Shows the processing used to create response messages that contain all or part of the routing table. |
| version-2/output-processing/ns1 | Shows the processing used to create response messages that contain all or part of the routing table. |
| version-2/response-processing/ns1 | Shows the processing of response messages. |
| version-2/response-processing/ns3 | Shows the processing of response messages. |
| version-2/split-horizon/ns1 | Shows the split horizon mechanism. |
| version-2/split-horizon/ns2 | Shows the split horizon mechanism. |

4.4 Static and Default Routes

4.4.1 Description

QualNet allows users to specify static routes that take precedence over those discovered by any routing protocols. These routes remain fixed throughout the simulation, even if network conditions (mobility, link failures) render the route useless, causing packet drops. There are also no route acquisition related overheads with static routes, such as packet transmissions, receptions, computational overhead for producing routes, or route acquisition delays that would normally be present using a traditional routing model. If no routing protocol is running on the node, and there is no static or default route to a given destination, corresponding packets will be dropped.

If a node is configured to use an on-demand routing protocol, then only routes discovered by the routing protocol are used even if static and/or default routes are enabled at the node.

If a node is configured to use a proactive routing protocol and static routes are enabled at the node, then the static routes take precedence over routes discovered by the routing protocol.

If static routes are disabled at a node and the node is not configured to use any routing protocol, then default routes are used.

4.4.2 Command Line Configuration

4.4.2.1 Static Routes

To enable static routes, include the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] STATIC-ROUTE YES
```

The scope of this parameter declaration can be Global or Node. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

Note: By default, static routes are not enabled.

Static Route Parameters

[Table 4-16](#) lists the static route parameters specified in the scenario configuration (.config) file. See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 4-16. Static Route Parameters

| Parameters | Value | Description |
|---|----------|---|
| STATIC-ROUTE-FILE <i>Required</i> Scope: Global, Node | Filename | Name of the static route file. The format of the static route file is described in Section 4.4.2.3 . |

4.4.2.2 Default Routes

To enable default routes, include the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] DEFAULT-ROUTE YES
```

The scope of this parameter declaration can be Global or Node. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

Note: By default, default routes are not enabled.

Default Route Parameters

[Table 4-17](#) lists the default route parameters specified in the scenario configuration (.config) file. See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 4-17. Default Route Parameters

| Parameters | Value | Description |
|--|----------|---|
| DEFAULT-ROUTE-FILE <i>Required</i> Scope: Global, Node | Filename | Name of the default route file. The format of the default route file is described in Section 4.4.2.3 . |

4.4.2.3 Format of Static and Default Route Files

Both the static route file and the default route file have the same format. Each line in either of these files has the following syntax:

<Node ID> <Destination Address> <Next Hop>

where

- | | |
|-----------------------|---|
| <Node ID> | Node identifier. |
| <Destination Address> | Destination address. This can be either a host IP address or a network IP address. |
| <Next Hop> | IP address of the next hop. |

Examples

1. The following is an example of a static/default route file for an IPv4 network. In the following example, packets from node 1 to the networks 0.0.2.0 and 0.0.3.0 use node 1's outgoing interface identified by the IP Address 0.0.1.2.

```
1 N2-2.0 1.2
1 N2-3.0 1.2
```

2. The following is an example of a static/default route file for an IPv6 network. In the following example, packets from node 1 to the networks TLA-0.NLA-0.SLA-2 and TLA-0.NLA-0.SLA-3 use node 1's outgoing interface identified by the IPv6 Address 2000::1:0:0:0:2.

```
1 TLA-0.NLA-0.SLA-2 2000::1:0:0:0:2
1 TLA-0.NLA-0.SLA-3 2000::1:0:0:0:2
```

4.4.3 GUI Configuration

This section describes how to configure static and default routes in the GUI.

5.1.5.1 Configuring Static Routes

To configure static routes, perform the following steps:

1. Go to **Default Device Properties Editor > Node Configuration > Routing Protocol**.
2. Set **Specify Static Routes** to Yes, and set the dependant parameters listed in [Table 4-18](#).

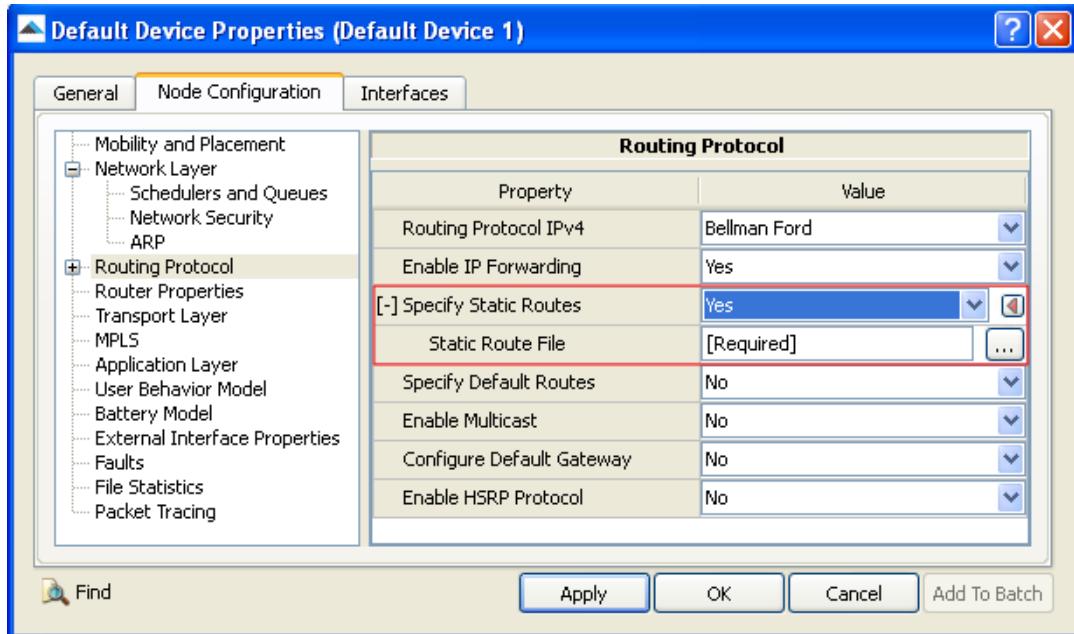


FIGURE 4-4. Configuring Static Routes

TABLE 4-18. Command Line Equivalent of Static Route Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-------------------|------------------------|------------------------|
| Static Route File | Node | STATIC-ROUTE-FILE |

Setting Parameters

- Set **Static Route File** to specify the name of the static route file. The format of the static route file is described in [Section 4.4.2.3](#).

5.1.5.2 Configuring Default Routes

To configure default routes, perform the following steps:

1. Go to **Default Device Properties Editor > Node Configuration > Routing Protocol**.
2. Set **Specify Default Routes** to Yes, and set the dependant parameters listed in [Table 4-19](#).

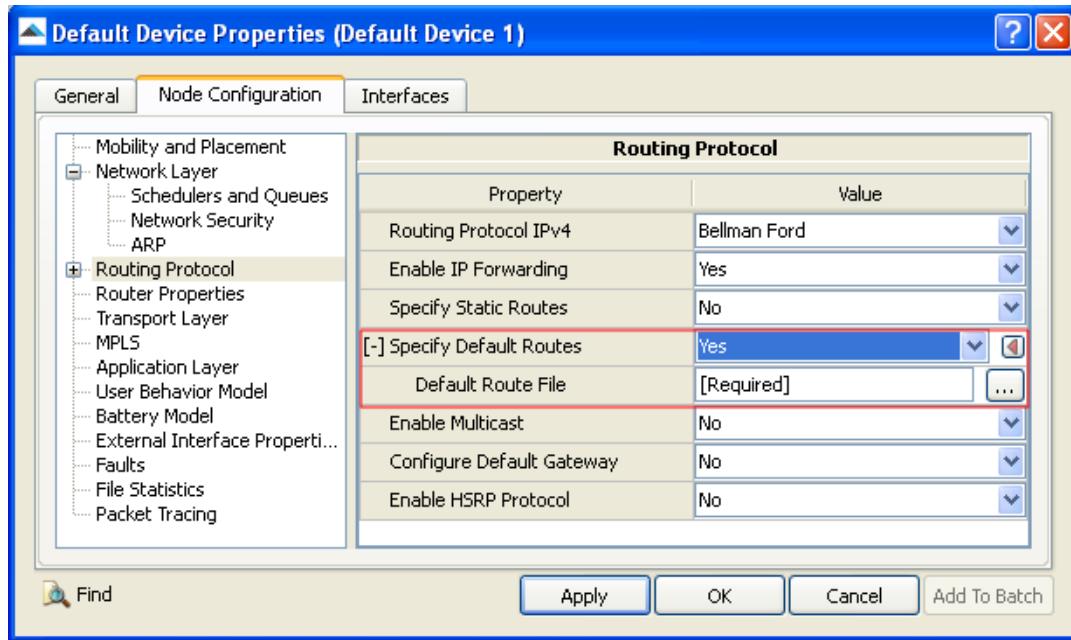


FIGURE 4-5. Configuring Default Routes

TABLE 4-19. Command Line Equivalent of Default Route Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|--------------------|------------------------|------------------------|
| Default Route File | Node | DEFAULT-ROUTE-FILE |

Setting Parameters

- Set *Default Route File* to specify the name of the static route file. The format of the default route file is described in [Section 4.4.2.3](#).

4.4.4 Scenarios Included in QualNet

The QualNet distribution includes several sample scenarios for Static and Default routes. All scenarios are located in the directory `QUALNET_HOME/scenarios/developer/dualip/gui-sample`. [Table 4-20](#) lists the sub-directory where each scenario is located.

TABLE 4-20. Static and Default Routing Scenarios Included in QualNet

| Scenario | Description |
|------------|--|
| gui-sample | Shows the functionality of static and default routing. |

5

Multicast Routing Protocol Models

This chapter describes features, configuration requirements and parameters, statistics, and scenarios for Multicast Routing Protocol Models, and consists of the following sections:

- Static Multicast Routes

5.1 Static Multicast Routes

5.1.1 Description

Static multicast routes are user-configured multicast routes. User can configure these routes in multicast static route file. QualNet Static Multicast Routes model supports both IPv4 and IPv6.

5.1.2 Command Line Configuration

To enable static multicast routes, include the following parameter in the scenario configuration (.config) file:

| | |
|--------------------------------------|-----|
| [<Qualifier>] MULTICAST-STATIC-ROUTE | YES |
|--------------------------------------|-----|

The scope of this parameter declaration can be Global or Node. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

Note: By default, static multicast routes are not enabled.

Static Multicast Routes Parameters

[Table 5-1](#) lists the static multicast routes parameters. See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 5-1. Static Multicast Routes Parameters

| Parameters | Value | Description |
|---|----------|---|
| MULTICAST-STATIC-ROUTE-FILE <i>Required</i> Scope: Global, Node | Filename | Name of the multicast static route file. The format of the static route file is described in Section 5.1.2.1 . |

5.1.2.1 Format of the Static Multicast Route File

Each line of the Static Multicast Route file has the following format:

<node ID> <source-address> <multicast-address> <interface-addresses>

where:

- <node ID> Node ID.
- <source-address> Source address.
- <multicast-address> Destination multicast group address.
- <interface-addresses> List of space-separated outgoing interface addresses.

Examples

1. The following is an example of a Static Multicast Route file for an IPv4 network. Node 1 will forward each multicast packet from source 192.168.0.1 to multicast group destination 225.0.0.1 on outgoing interface 192.168.0.1. Node 2 will forward each multicast packet from source 192.168.0.1 to multicast group destination 225.0.0.1 on outgoing interfaces 192.168.0.2 and 192.168.1.2.

```
1 192.168.0.1 225.0.0.1 192.168.0.1
2 192.168.0.1 225.0.0.1 192.168.0.2 192.168.1.2
```

2. The following is an example of a Static Multicast Route file for an IPv6 network. Node 1 will forward each multicast packet from source 1000:1::1 to multicast group destination ff12::3 on outgoing interface 1000:1::1. Node 2 will forward each multicast packet from source 1000:1::1 to multicast group destination ff12::3 on outgoing interfaces 1000:2::1 and 1000:5::1.

```
1 1000:1::1 ff12::3 1000:1::1
2 1000:1::1 ff12::3 1000:2::1 1000:5::1
```

5.1.3 GUI Configuration

To configure static multicast routes for a particular node, perform the following steps:

1. Go to **Default Device Properties Editor > Node Configuration > Routing Protocol**.
2. Set **Enable Multicast** to Yes.

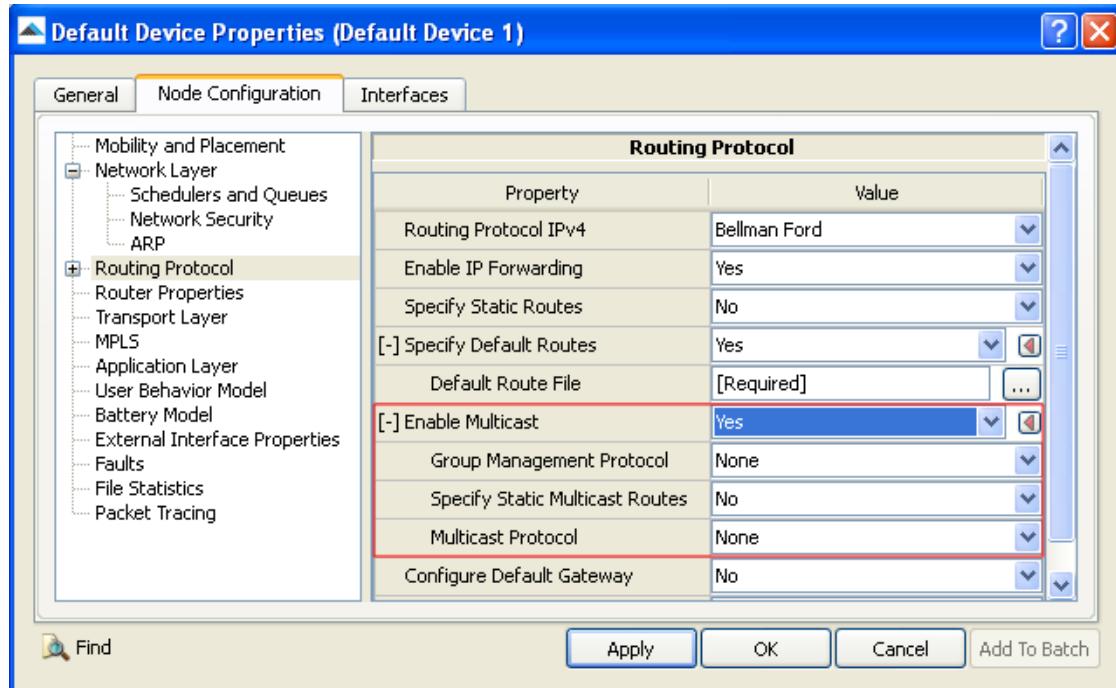


FIGURE 5-1. Enabling Multicast

3. Set **Specify Static Multicast Routes** to Yes, and set the dependent parameters listed in [Table 5-2](#).

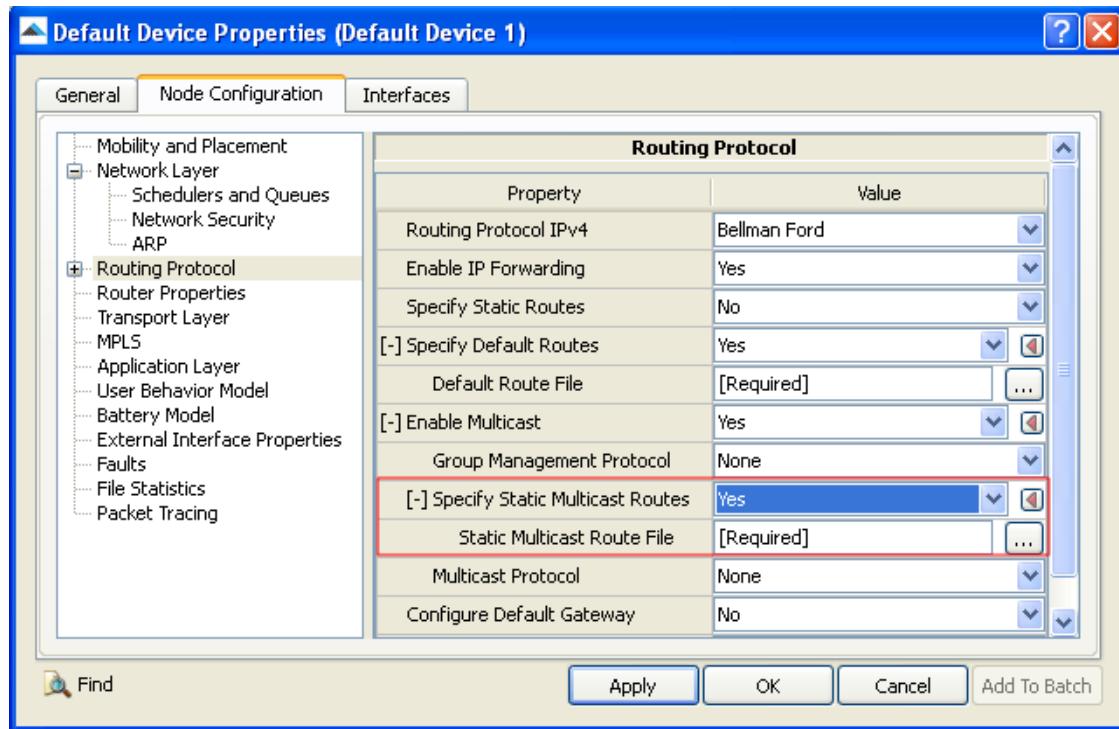


FIGURE 5-2. Configuring Static Multicast Routes

TABLE 5-2. Command Line Equivalent of Static Multicast Route Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-----------------------------|------------------------|-----------------------------|
| Static Multicast Route File | Node | MULTICAST-STATIC-ROUTE-FILE |

Setting Parameters

- Set **Static Multicast Route File** to the name of the Static Multicast Route file. The format of the multicast static route file is described in [Section 5.1.2.1](#).

5.1.4 Statistics

There are no statistics generated for Static Multicast routes.

5.1.5 Scenarios Included in QualNet

The QualNet distribution includes several sample scenarios for Static Multicast routes. All scenarios are located in the directory `QUALNET_HOME/scenarios/developer/ipv6`. [Table 5-3](#) lists the sub-directory where each scenario is located.

TABLE 5-3. Static Multicast Route Scenarios Included in QualNet

| Scenario | Description |
|------------------|--|
| static-multicast | Shows the functionality of Static-multicast in IPv6 network. |

6 Queues and Scheduler Models

This chapter describes features, configuration requirements and parameters, statistics, and scenarios for Queues and Scheduler Models, and consists of the following sections:

- First-In First-Out (FIFO) Queue
- Random Early Detection (RED) Queue
- Random Early Detection with In/Out (RIO) Queue
- Round Robin Scheduler
- Self-Clocked Fair Queueing (SCFQ) Scheduler
- Strict Priority Scheduler
- Weighted Fair Queueing (WFQ) Scheduler
- Weighted RED (WRED) Queue
- Weighted Round Robin (WRR) Scheduler

6.1 Class-Based Queuing (CBQ)

6.1.1 Description

CBQ is a link-sharing resource manager used to impose specific rules for the distribution of bandwidth among different agencies sharing the same physical link, and provide *assured bandwidth* service. CBQ controls the allocation of resources on a per-hop basis, as well as on an end-to-end (per flow) basis.

6.1.2 Features and Assumptions

This section describes the implemented features, omitted features, assumptions and limitations of the CBQ model.

6.1.2.1 Implemented Features

- Each class has its own queue and is assigned its share of bandwidth. A child class can borrow bandwidth from its parent class as long as excess bandwidth is available.

6.1.2.2 Omitted Features

None.

6.1.2.3 Assumptions and Limitations

- The root class is allocated 100% of the link bandwidth, and for each non-leaf class, the sum of the bandwidth shares allocated to child classes equals the bandwidth allocated to the class itself.
- The agencies just below the ROOT must consist of all non-leaf agencies.
- There could be only one queue with a particular priority value.
- Leaf Classes or Agencies are labelled "1". If the TOP-LEVEL specified is greater than maximum level in the link sharing structure, it will then automatically search up to the ROOT level.

6.1.3 Command Line Configuration

To select the CBQ scheduler, include the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] IP-QUEUE-SCHEDULER           CBQ
```

The scope of this parameter declaration can be Global, Node, Subnet, or Interface. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

CBQ Parameters

Table 6-1 shows the CBQ scheduler parameters. See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 6-1. CBQ Parameters

| Parameter | Values | Description |
|--|--|--|
| LINK-SHARING-STRUCTURE-FILE <i>Required</i> Scope: All | Filename | Name of the Link Sharing Structure file containing configuration information about weights, priorities, and link sharing between agencies. Per hop and end-to-end (per flow) link sharing and allocation of resources is specified in this file. The format of the Link Sharing Structure file is described in Table 6.1.3.1 . |
| CBQ-GENERAL-SCHEDULER <i>Required</i> Scope: All | List: • PRR • WRR | Packet scheduler that CBQ uses to manage the queues. PRR : Prioritized Round Robin scheduling protocol. WRR : Weighted Round Robin scheduling protocol. |
| CBQ-LINK-SHARING-GUIDELINE <i>Required</i> Scope: All | List: • ANCESTOR-ONLY • TOP-LEVEL | Determines whether or not bandwidth is regulated by the link sharing scheduler. Bandwidth remains unregulated in the ANCESTOR-ONLY case when the class of traffic is under-limit, or its immediate ancestor is under-limit. Bandwidth remains unregulated in the TOP-LEVEL case when the class of traffic is under-limit, or at least one of its ancestors up to CBQ-TOP-LEVEL generations above, are under-limit. |
| CBQ-TOP-LEVEL <i>Required</i> Scope: All | Integer: <i>Range:</i> ≥ 1 | This parameter determines the maximum number of generations to search, for under-limit ancestors. If the TOP-LEVEL specified is greater than or equal to the maximum level in the link-sharing structure, then it will search up to the ROOT level. |
| INPUT-SCHEDULER-STATISTICS <i>Optional</i> Scope: All | List: • YES • NO <i>Default:</i> NO | Indicates whether statistics are collected for input schedulers. |
| SCHEDULER-STATISTICS <i>Optional</i> Scope: All | List: • YES • NO <i>Default:</i> NO | Indicates whether statistics are collected for output schedulers. |

6.1.3.1 Format of the Link Sharing Structure File

The Link Sharing Structure file contains configuration information about weights, priorities, and link sharing between agencies. Per-hop and end-to-end (per flow) link sharing and allocation of resources is specified in this file.

Each line in the Link Sharing Structure file has the following format:

```
<Node> <Interface> <Agency>, <Descendant-1>, . . . , <Descendant-N>
```

where

| | |
|----------------|--|
| <Node> | Node ID of the node for which this specification is valid, or ANY for all nodes. |
| <Interface> | IP address of the network interface for which this specification is valid, or ANY for all network interfaces on the node(s). |
| <Agency> | Name of the root agency. |
| <Descendant-i> | Descendant specification. A descendant can be a non-leaf agency, a leaf agency, or an application. The format for specifying the different types of descendants are described below. |

A non-leaf agency is specified using the following format:

```
<Agency> <Weight> <Borrow> <Efficient>
```

A leaf agency or application is specified using the following format:

```
<Priority> <Weight> <Borrow> <Efficient>
```

where

| | |
|-------------|---|
| <Agency> | Name of the agency. |
| <Priority> | Priority of the agency. This value is an integer. The default value is -1. |
| <Weight> | Weight for the agency. This value is a real number. The default value is 1.0 |
| <Borrow> | Specifies whether the agency is allowed to borrow or not. This value can be True or False. The default value is FALSE |
| <Efficient> | Specifies whether or not the agency is efficient. This value can be True or False. The default value is FALSE |

The default values are used to assign values to the parameters of the root node. The values of all the other nodes are input by the user. The parameters cannot be omitted. Hence, the agency definition should have all the required fields.

6.1.4 GUI Configuration

To configure the CBQ in the GUI, perform the following steps:

1. Go to one of the following locations:
 - To set wireless subnet properties, go to **Wireless Subnet Properties Editor > Network Layer > Schedulers and Queues**.
 - To set properties for a specific node, go to **Node Properties Editor > Node Configuration > Network Layer > Schedulers and Queues**.
 - To set properties for a specific interface of a node, go to one of the following locations:
 - **Interface Properties Editor > Interfaces > Interface # > Network Layer > Schedulers and Queues**
 - **Default Device Properties Editor > Interfaces > Interface # > Network Layer > Schedulers and Queues**.

In this section, we show how to configure CBQ parameters for a specific node using the Default Device Properties Editor. Parameters can be set in the other properties editors in a similar way.

2. Set **IP Output Queue Scheduler** to *Class Based Queueing* and set the dependent parameters listed in [Table 6-2](#).

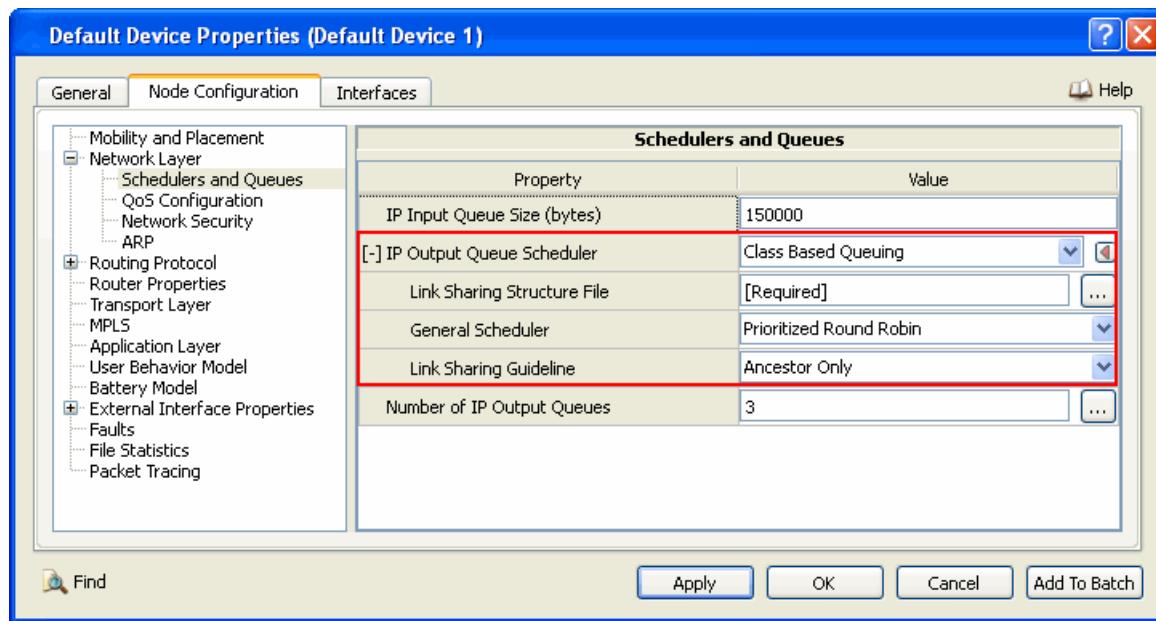


FIGURE 6-1. Setting CBQ Parameters

TABLE 6-2. Command Line Equivalent of CBQ Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-----------------------------|-------------------------|-----------------------------|
| Link sharing Structure File | Node, Subnet, Interface | LINK-SHARING-STRUCTURE-FILE |
| General Scheduler | Node, Subnet, Interface | CBQ-GENERAL-SCHEDULER |
| Link Sharing Guideline | Node, Subnet, Interface | CBQ-LINK-SHARING-GUIDELINE |

Setting Parameters

- Set **Link Sharing Structure File** to the name of the CBQ configuration file. The format of the CBQ configuration file is described in [Section 6.1.3.1](#).
- To set CBQ Top Level, set **Link Sharing Guideline** to *Top Level*; otherwise, set **Link Sharing Guideline** to *Ancestor Only*.

3. If **Link Sharing Guideline** is set to *Top Level*, then set the dependent parameters listed in [Table 6-3](#).

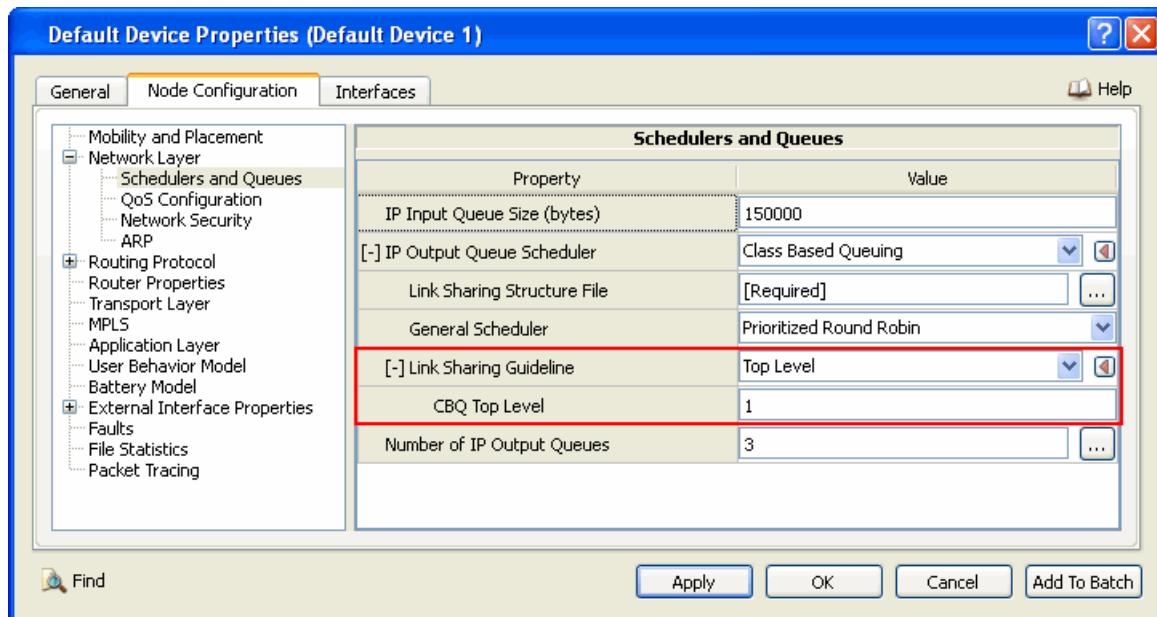


FIGURE 6-2. Specify Link Sharing Guideline

TABLE 6-3. Command Line Equivalent of Link Sharing Guideline Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|-------------------------|------------------------|
| CBQ Top Level | Node, Subnet, Interface | CBQ-TOP-LEVEL |

Configuring Statistics Parameters

Statistics for the CBQ scheduler can be collected at the global, node, subnet, and interface levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for IP Input Queue Scheduler, check the box labeled **IP Input Scheduler** in the appropriate properties editor.

To enable statistics collection for IP Output Queue Scheduler, check the box labeled **IP Output Scheduler** in the appropriate properties editor.

TABLE 6-4. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------------|---------------------------------|----------------------------|
| IP Input Scheduler | Global, Node, Subnet, Interface | INPUT-SCHEDULER-STATISTICS |
| IP Output Scheduler | Global, Node, Subnet, Interface | SCHEDULER-STATISTICS |

6.1.5 Statistics

Table 6-5 lists the CBQ statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 6-5. CBQ Statistics

| Statistic | Description |
|----------------------------|--|
| Packets Queued | Total number of packets enqueued at the interface |
| Packets Dequeued | Total number of packets dequeued at the interface |
| Number of Dequeue Requests | Total number of dequeue requests received. |
| Selection | Number of dequeue packets selected on the basis of General scheduler, link-sharing scheduler and forced packets. |
| Number of times Punished | Total number of times the packets were punished or suspended. |
| Max Extradelay | Maximum value of delay occurred due to packets punished/suspended (seconds). |
| Avg Extradelay | Average value of delay occurred due to packets punished/suspended (seconds). |

6.1.6 References

1. Sally Floyd and Van Jacobson, "Link-sharing and Resource Management Models for Packet Networks", *IEEE/ACM Transactions on Networking*, Vol 3, No. 4, Aug 1995.

6.2 First-In First-Out (FIFO) Queue

6.2.1 Description

FIFO queue accepts packets until it is full (the total size, in bytes, of all the packets in the queue determine whether the queue is full or not). Once the queue is full, all packets that arrive at the queue are dropped, until packets are dequeued, and space becomes available.

6.2.2 Command Line Configuration

To specify FIFO as the queue discipline, include the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] IP-QUEUE-TYPE [<Index>] FIFO
```

where

<Index> Queue index to which this parameter declaration is applicable, enclosed in square brackets. This should be in the range 0 to $n - 1$, where n is the number of priority queues at the interface.

The instance specification is optional. If an instance is not included, then the parameter declaration is applicable to all queues at the interface

The scope of this parameter declaration can be Global, Node, Subnet, or Interface. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

FIFO Parameters

[Table 6-6](#) shows the FIFO configuration parameters. See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 6-6. FIFO Parameters

| Parameter | Value | Description |
|--|---|-------------|
| IP-QUEUE-PRIORITY-QUEUE-SIZE <i>Optional</i> Scope: All <i>Instances: queue index</i> | Integer <i>Range: > 0</i> <i>Default: 150000</i> <i>Unit: bytes</i> | Queue size. |

TABLE 6-6. FIFO Parameters (Continued)

| Parameter | Value | Description |
|---|--|---|
| INPUT-QUEUE-STATISTICS <i>Optional</i> Scope: All | List: • YES • NO <i>Default:</i> NO | Indicates whether statistics are collected for input queues. |
| QUEUE-STATISTICS <i>Optional</i> Scope: All | List: • YES • NO <i>Default:</i> NO | Indicates whether statistics are collected for output queues. |

6.2.3 GUI Configuration

This section describes how to configure FIFO in the GUI.

Configuring FIFO Parameters

To configure the FIFO parameters, perform the following steps:

1. Go to one of the following locations:
 - To set wireless subnet properties, go to **Wireless Subnet Properties Editor > Network Layer > Schedulers and Queues**.
 - To set properties for a specific node, go to **Node Properties Editor > Node Configuration > Network Layer > Schedulers and Queues**.
 - To set properties for a specific interface of a node, go to one of the following locations:
 - **Interface Properties Editor > Interfaces > Interface # > Network Layer > Schedulers and Queues**
 - **Default Device Properties Editor > Interfaces > Interface # > Network Layer > Schedulers and Queues**.

In this section, we show how to configure FIFO parameters for a specific node using the Default Device Properties Editor. Parameters can be set in the other properties editors in a similar way.

2. Set **Number of IP Output Queues** to the desired value.

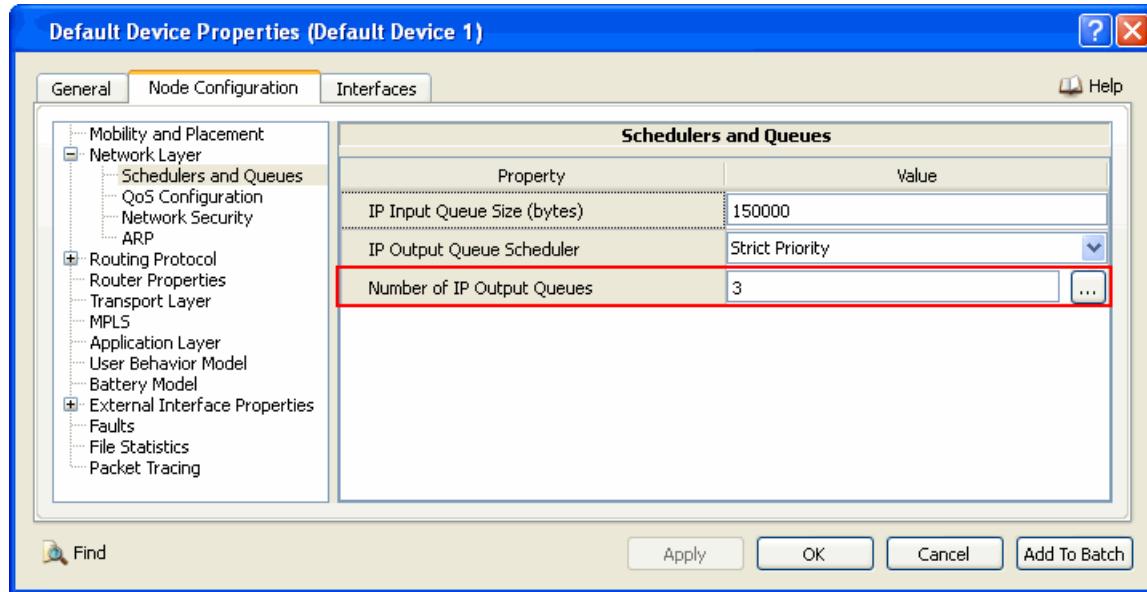


FIGURE 6-3. Setting Number of Queues

3. Click on the **Open Array Editor** button in the **Value** column. This opens the Array Editor.
4. In the left panel of the Array Editor, select the index of the queue to be configured. In the right panel, set **IP Queue Type** to *FIFO* and set the dependent parameters listed in [Table 6-7](#).

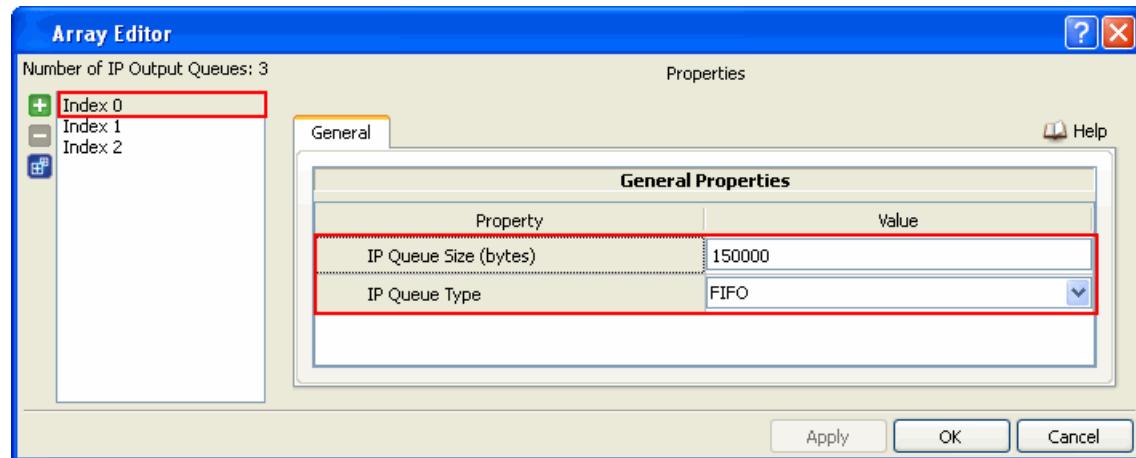


FIGURE 6-4. Setting FIFO Queue Parameters

TABLE 6-7. Command Line Equivalent of FIFO Queue Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|----------------------|-------------------------------|-------------------------------|
| IP Queue Size | Node, Interface, Subnet | IP-QUEUE-PRIORITY-QUEUE-SIZE |

Configuring Statistics Parameters

Statistics for FIFO can be collected at the global, node, subnet, and interface levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for IP input queues, check the box labeled **IP Input Queue** in the appropriate properties editor.

To enable statistics collection for IP output queues, check the box labeled **IP Output Queue** in the appropriate properties editor.

TABLE 6-8. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|----------------------|---------------------------------|-------------------------------|
| IP Input Queue | Global, Node, Subnet, Interface | INPUT-QUEUE-SATISTICS |
| IP Output Queue | Global, Node, Subnet, Interface | QUEUE-SATISTICS |

6.2.4 Statistics

Table 6-9 lists the FIFO statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 6-9. FIFO Statistics

| Statistic | Description |
|----------------------------------|---|
| Total Packets Queued | Total number of packets queued |
| Total Packets Dequeued | Total number of packets dequeued |
| Total Packets Dropped | Total number of packets dropped due to the queue being full |
| Total Packets Dropped Forcefully | Total number of packets dropped forcefully due to reasons other than the queue being full, such as the next hop not being reachable |
| Average Queue Length (bytes) | Average queue length in bytes (see note below) |
| Average Time in Queue | Average time spent by packets in the queue (in seconds) |
| Longest Time in Queue | Longest time spent in queue by a packet (in seconds) |
| Peak Queue Size (bytes) | Largest size ever reached by the queue (bytes) |

Note: The average queue length is calculated as follows:

$$\text{average queue length} = \frac{\text{sum of weighted packet sizes for all packets}}{\text{(simulation time, in seconds)}},$$

where

$$\text{weighted packet size} = (\text{packet size, in bytes}) * (\text{time spent by the packet in the queue, in seconds})$$

6.2.5 Scenarios Included in QualNet

The QualNet distribution includes several sample scenarios for the FIFO queue. All scenarios are located in the directory `QUALNET_HOME/scenarios/developer/queue/fifo`. [Table 6-10](#) lists the sub-directory where each scenario is located.

TABLE 6-10. FIFO Scenarios Included in QualNet

| Scenario | Description |
|----------|--------------------------------------|
| FIFO | Shows the performance of FIFO queue. |

6.3 Random Early Detection (RED) Queue

The QualNet RED Queue model is based on the following paper:

Sally Floyd and Van Jacobson, "Random Early Detection For Congestion Avoidance", *IEEE/ACM Transactions on Networking*, August 1993.

6.3.1 Description

The Random Early Detection (RED) queue drops packets at the current router based on drop probabilities, rather than at a subsequent congested router, if Explicit Congestion Notification (ECN) is off. (ECN is off by default.) If ECN is on, packets are marked instead of being dropped, and the source of the TCP traffic throttles back based on the notification.

RED is a congestion-avoidance algorithm that calculates average queue sizes in order to address incipient congestion. The algorithm also attempts to address global synchronization as well as maintaining fairness across data flows, without bias against bursty flows.

The possibility of random early drops begins when the RED queue averaging algorithm determines that the average queue size (i.e., the number of packets in the queue) is greater than or equal to the minimum threshold and less than the maximum threshold. The queue weight determines the bias towards recent or historical queue lengths in calculating the average queue size. The algorithm uses the time it would take to transmit a typical small packet to estimate the queue average during idle periods (when the queue is completely empty).

6.3.2 Features and Assumptions

This section describes the implemented features, omitted features, assumptions and limitations of the RED queue model.

6.3.2.1 Implemented Features

- When average queue size goes above maximum threshold, packets are dropped.
- When ECN is enabled, then packets are marked instead of being dropped else they are dropped.

6.3.2.2 Omitted Features

None.

6.3.2.3 Assumptions and Limitations

None.

6.3.3 Command Line Configuration

To specify RED as the queue discipline, include the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] IP-QUEUE-TYPE [<Index>] RED
```

where

| | |
|----------------------------|--|
| <code><Index></code> | <p>Queue index to which this parameter declaration is applicable, enclosed in square brackets. This should be in the range 0 to $n - 1$, where n is the number of priority queues at the interface.</p> <p>The instance specification is optional. If an instance is not included, then the parameter declaration is applicable to all queues at the interface</p> |
|----------------------------|--|

The scope of this parameter declaration can be Global, Node, Subnet, or Interface. See [Section 1.2.1.1](#) for a description of `<Qualifier>` for each scope.

RED Parameters

[Table 6-11](#) lists the configuration parameters for the RED queue model. See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 6-11. RED Parameters

| Parameter | Value | Description |
|--|--|---|
| IP-QUEUE-PRIORITY-QUEUE-SIZE <i>Optional</i> <i>Scope:</i> All <i>Instances:</i> queue index | Integer <i>Range:</i> > 0 <i>Default:</i> 150000 <i>Unit:</i> bytes | Queue size. |
| RED-MIN-THRESHOLD <i>Optional</i> <i>Scope:</i> All <i>Instances:</i> queue index | Integer <i>Range:</i> > 0 <i>Default:</i> 5 <i>Unit:</i> packets | Minimum threshold. If the queue size is equal to or larger than this number, packets can be randomly dropped. |
| RED-MAX-THRESHOLD <i>Optional</i> <i>Scope:</i> All <i>Instances:</i> queue index | Integer <i>Range:</i> > 0 <i>Default:</i> 15 <i>Unit:</i> packets | Maximum threshold. If the queue size is less than this number, packets can be randomly dropped. |
| RED-MAX-PROBABILITY <i>Optional</i> <i>Scope:</i> All <i>Instances:</i> queue index | Real <i>Range:</i> [0.0, 1.0] <i>Default:</i> 0.02 | Maximum probability of a packet being dropped. |
| RED-QUEUE-WEIGHT <i>Optional</i> <i>Scope:</i> All <i>Instances:</i> queue index | Real <i>Range:</i> [0.0, 1.0] <i>Default:</i> 0.002 | Queue weight . This is used to determine the bias towards recent queue lengths in calculating the average. The larger this value, greater is the bias towards recent queues. |

TABLE 6-11. RED Parameters (Continued)

| Parameter | Value | Description |
|--|--|---|
| RED-SMALL-PACKET-TRANSMISSION-TIME Optional Scope: All Instances: queue index | Time <i>Range:</i> > 0S <i>Default:</i> 10MS | Typical time to transmit a small packet. This is used to estimate the queue average during idle periods. |
| INPUT-QUEUE-STATISTICS Optional Scope: All | List: • YES • NO <i>Default:</i> NO | Indicates whether statistics are collected for input queue. |
| QUEUE-STATISTICS Optional Scope: All | List: • YES • NO <i>Default:</i> NO | Indicates whether statistics are collected for output queue. |

6.3.4 GUI Configuration

This section describes how to configure RED in the GUI.

Configuring RED Parameters

To configure the RED parameters, perform the following steps:

1. Go to one of the following locations:
 - To set wireless subnet properties, go to **Wireless Subnet Properties Editor** > **Network Layer** > **Schedulers and Queues**.
 - To set properties for a specific node, go to **Node Properties Editor** > **Node Configuration** > **Network Layer** > **Schedulers and Queues**.
 - To set properties for a specific interface of a node, go to one of the following locations:
 - **Interface Properties Editor** > **Interfaces** > **Interface #** > **Network Layer** > **Schedulers and Queues**
 - **Default Device Properties Editor** > **Interfaces** > **Interface #** > **Network Layer** > **Schedulers and Queues**.

In this section, we show how to configure RED parameters for a specific node using the Default Device Properties Editor. Parameters can be set in the other properties editors in a similar way.

2. Set **Number of IP Output Queues** to the desired value.

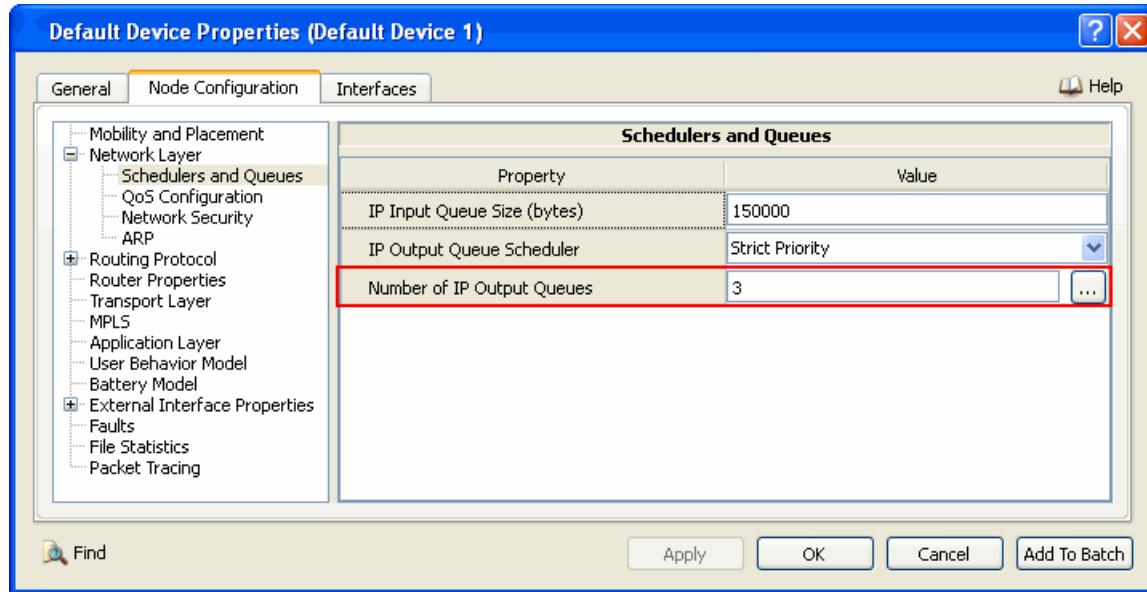


FIGURE 6-5. Setting Number of Queues

3. Click on the **Open Array Editor** [...] button in the **Value** column. This opens the Array Editor.
4. In the left panel of the Array Editor, select the index of the queue to be configured. In the right panel, set **IP Queue Type** to *RED* and set the dependent parameters listed in Table 6-12.

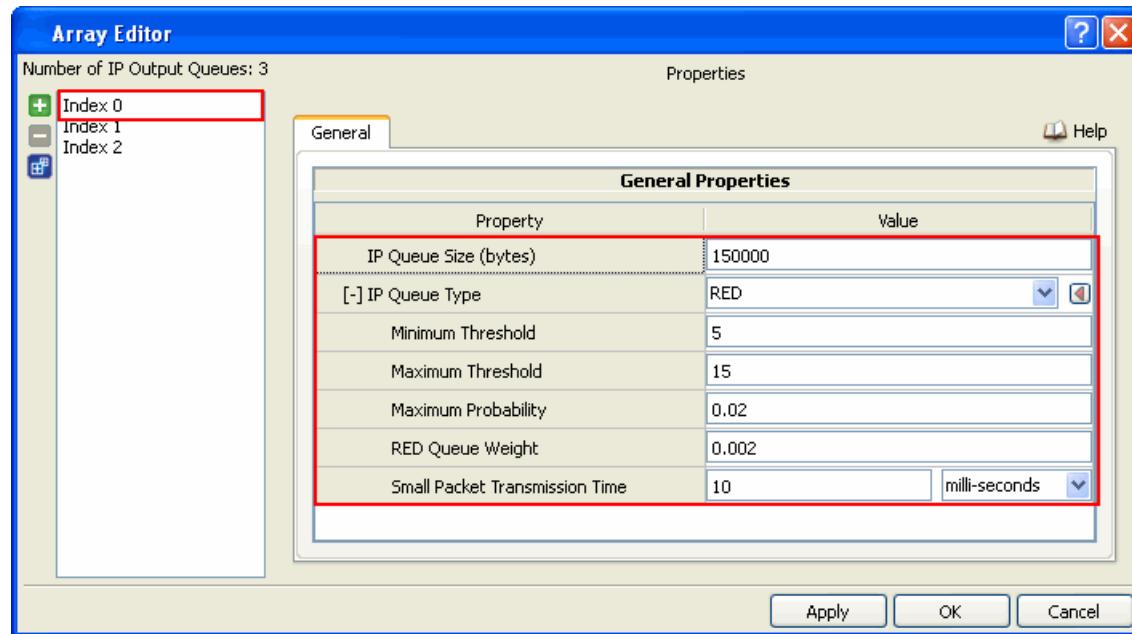


FIGURE 6-6. Setting RED Queue Parameters

TABLE 6-12. Command Line Equivalent of RED Queue Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|--------------------------------|-------------------------|------------------------------------|
| IP Queue Size | Node, Subnet, Interface | IP-QUEUE-PRIORITY-QUEUE-SIZE |
| Minimum Threshold | Node, Subnet, Interface | RED-MIN-THRESHOLD |
| Maximum Threshold | Node, Subnet, Interface | RED-MAX-THRESHOLD |
| Maximum Probability | Node, Subnet, Interface | RED-MAX-PROBABILITY |
| RED Queue Weight | Node, Subnet, Interface | RED-QUEUE-WEIGHT |
| Small Packet Transmission Time | Node, Subnet, Interface | RED-SMALL-PACKET-TRANSMISSION-TIME |

Configuring Statistics Parameters

Statistics for RED can be collected at the global, node, subnet, and interface levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for IP input queues, check the box labeled **IP Input Queue** in the appropriate properties editor.

To enable statistics collection for IP output queues, check the box labeled **IP Output Queue** in the appropriate properties editor.

TABLE 6-13. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-----------------|---------------------------------|------------------------|
| IP Input Queue | Global, Node, Subnet, Interface | INPUT-QUEUE-SATISTICS |
| IP Output Queue | Global, Node, Subnet, Interface | QUEUE-SATISTICS |

6.3.5 Statistics

Table 6-14 lists the RED queue statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 6-14. RED Statistics

| Statistic | Description |
|----------------------------------|--|
| Total Packets Queued | Total number of packets queued |
| Total Packets Dequeued | Total number of packets dequeued |
| Total Packets Dropped | Total number of packets dropped due to the queue being full |
| Average Queue Length (bytes) | Average queue length (bytes). See note below. |
| Average Time in Queue | Average time spent by packets in the queue (seconds) |
| Longest Time in Queue | Longest time spent in queue by a packet (seconds) |
| Peak Queue Size (bytes) | Largest size ever reached by the queue (bytes) |
| Total Packets Dropped Forcefully | Total packets dropped forcefully due to reasons other than the queue being full, such as the next hop not being reachable. |
| Packets Marked ECN | Total number of packets marked by ECN capable network |
| Longest Time in Queue | Longest delay suffered by a packet in this queue (seconds) |

Note: The average queue length is calculated as follows:

$$\text{average queue length} = \text{sum of weighted packet sizes for all packets} / (\text{simulation time, in seconds}),$$

where

$$\text{weighted packet size} = (\text{packet size, in bytes}) * (\text{time spent by the packet in the queue, in seconds})$$

6.3.6 Scenarios Included in QualNet

The QualNet distribution includes several sample scenarios for the RED queue. All scenarios are located in the directory `QUALNET_HOME/scenarios/developer`. [Table 6-15](#) lists the sub-directory where each scenario is located.

TABLE 6-15. RED Scenarios Included in QualNet

| Scenario | Description |
|-------------------------|---|
| queue/red | Shows how packets are handled by RED queue when bottleneck condition arrives. |
| ecn/red/red-with-ecn | Shows the behavior of the RED with ECN. |
| ecn/red/red-without-ecn | Shows the behavior of the RED without ECN. |

6.3.7 References

1. Sally Floyd and Van Jacobson, "Random Early Detection For Congestion Avoidance", *IEEE/ACM Transactions on Networking*, August 1993.

6.4 Random Early Detection with In/Out (RIO) Queue

6.4.1 Description

RIO is a multilevel RED queue management protocol (MRED) queuing algorithm suitable for the AF PHB that can identify any preferences and can punish misbehaving users. The incoming packets are marked through packet marking algorithm on the basis of Service Level Agreement (SLA) between the customer and service-provider. The marking is done in two-color (DP0 and DP1) or three-color (DP0, DP1, DP2) level referring drop precedence (DP) as below:

- Green: Level Low Drop, DP0
- Yellow: Level Medium Drop, DP1
- Red: Level High Drop, DP2

RIO is a Multiple Average Multiple threshold (MAMT) variant of multilevel RED and can operate in both two and three color modes. Two simple approaches to the MAMT variant of MRED are RIO-C (RED with In/Out and Coupled average Queues) and RIO-DC (RED with In/Out and Decoupled average Queues).

RIO-C derives its name from the coupled relationship of the average queue calculation. The average queue for packets of different colors can be calculated by adding its average queue to the average queues of colors of lower drop precedence. The implication of this scheme is that the drop probability for packets with higher drop precedence is dependent on the buffer occupancy of packets having lower drop precedence. Thus, there is an inherent assumption that it is better to drop DP1 and DP2 packets for DP0 packets. Similarly, for DP1 packets, it is assumed better to drop DP2 packets. Services based on AF implementations utilizing RIO-C will be subject to this restriction.

RIO-DC (RIO with Decoupled Queues), the average queue for packets of each color is calculated independently. In this scheme, the average queue length for a color is calculated using number of packets of that color in the queue. The RED parameter settings can be chosen depending on the treatment to be given to different colors. RIO-DC appears to be the most suitable MRED-variant if the operator desires to assign weights to different colored packets within the same queue i.e. there is no relationship between the packets marked green and yellow. Thus, the RIO-DC scheme could start to emulate the WRR scheduling scheme except that a single queue is used.

6.4.2 Command Line Configuration

To specify RIO as the queue discipline, include the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] IP-QUEUE-TYPE [<Index>] RIO
```

where

| | |
|---------|---|
| <Index> | Queue index to which this parameter declaration is applicable, enclosed in square brackets. This should be in the range 0 to $n - 1$, where n is the number of priority queues at the interface. |
|---------|---|

The instance specification is optional. If an instance is not included, then the parameter declaration is applicable to all queues at the interface

The scope of this parameter declaration can be Global, Node, Subnet, or Interface. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

RIO Parameters

RIO configuration parameters are described in [Table 6-16](#). See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 6-16. RIO Parameters

| Parameter | Value | Description |
|---|--|---|
| IP-QUEUE-PRIORITY-QUEUE-SIZE <i>Optional</i> Scope: All Instances: queue index | Integer <i>Range:</i> > 0 <i>Default:</i> 150000 <i>Unit:</i> bytes | Queue size. |
| RIO-COUNTING-MODE <i>Required</i> Scope: All Instances: queue index | List: • COUPLED • DECOUPLED | Specifies the relationship of the average queue calculation. Coupled mode counts Green profile packets towards the calculation for dropping Yellow profile packets. Decoupled mode counts Green and Yellow profile packets separately. |
| RIO-COLOR-MODE <i>Required</i> Scope: All Instances: queue index | List: • TWO-COLOR • THREE-COLOR | Specifies the treatment to be given to RIO-COLOR-MODE. In TWO-COLOR mode it uses twin RED algorithms: • the first for the Green profile packets and • the second for Yellow profile packets In THREE-COLOR mode it uses three RED algorithms: • the first for the Green profile packets and • the second for Yellow profile packets and • the third for Red profile packets. |
| GREEN-PROFILE-MIN-THRESHOLD <i>Optional</i> Scope: All Instances: queue index | Integer <i>Range:</i> > 0 <i>Default:</i> 10 <i>Unit:</i> packets | Specifies the number of green profile packets in the queue that represents the lower bound at which green packets can be randomly dropped. |
| GREEN-PROFILE-MAX-THRESHOLD <i>Optional</i> Scope: All | Integer <i>Range:</i> > 0 <i>Default:</i> 20 <i>Unit:</i> packets | Specifies the number of green profile packets in the queue that represents the upper bound at which green packets can be randomly dropped. |

TABLE 6-16. RIO Parameters

| Parameter | Value | Description |
|---|--|---|
| GREEN-PROFILE-MAX-PROBABILITY <i>Optional</i> Scope: All Instances: queue index | Real <i>Range:</i> [0.0, 1.0] <i>Default:</i> 0.02 | Specifies maximum probability value at which a packet can be dropped (before the queue is completely full). |
| YELLOW-PROFILE-MIN-THRESHOLD <i>Optional</i> Scope: All Instances: queue index | Integer <i>Range:</i> > 0 <i>Default:</i> 5 <i>Unit:</i> packets | Specifies the number of yellow profile packets in the queue that represents the lower bound at which green packets can be randomly dropped. |
| YELLOW-PROFILE-MAX-THRESHOLD <i>Optional</i> Scope: All Instances: queue index | Integer <i>Range:</i> > 0 <i>Default:</i> 10 <i>Unit:</i> packets | Specifies the number of yellow profile packets in the queue that represents the upper bound at which green packets can be randomly dropped. |
| YELLOW-PROFILE-MAX-PROBABILITY <i>Optional</i> Scope: All Instances: queue index | Real <i>Range:</i> [0.0, 1.0] <i>Default:</i> 0.02 | Specifies maximum probability value at which a packet can be dropped (before the queue is completely full). |
| RED-PROFILE-MIN-THRESHOLD <i>Optional</i> Scope: All Instances: queue index | Integer <i>Range:</i> > 0 <i>Default:</i> 2 <i>Unit:</i> packets | Specifies the number of red profile packets in the queue that represents the lower bound at which green packets can be randomly dropped. |
| RED-PROFILE-MAX-THRESHOLD <i>Optional</i> Scope: All Instances: queue index | Integer <i>Range:</i> > 0 <i>Default:</i> 5 <i>Unit:</i> Packets | Specifies the number of red profile packets in the queue that represents the upper bound at which green packets can be randomly dropped. |
| RED-PROFILE-MAX-PROBABILITY <i>Optional</i> Scope: All Instances: queue index | Real <i>Range:</i> [0.0, 1.0] <i>Default:</i> 0.02 | Specifies maximum probability that a packet can be dropped (before the queue is completely full). |

TABLE 6-16. RIO Parameters

| Parameter | Value | Description |
|--|--|--|
| RED-QUEUE-WEIGHT <i>Optional</i> Scope: All <i>Instances:</i> queue index | Real <i>Range:</i> > 0 . 0 <i>Default:</i> 0 . 002 | Specifies the queue weight used to determine the bias towards recent or historical queue lengths in calculating the average. |
| RED-SMALL-PACKET-TRANSMISSION-TIME <i>Optional</i> Scope: All <i>Instances:</i> queue index | Time <i>Range:</i> > 0 S <i>Default:</i> 10MS | Specifies sample amount of time to transmit a small packet. Used to estimate the queue average during idle periods. |
| INPUT-QUEUE-STATISTICS <i>Optional</i> Scope: All | List: • YES • NO <i>Default:</i> NO | Indicates whether statistics are collected for input queue. |
| QUEUE-STATISTICS <i>Optional</i> Scope: All | List: • YES • NO <i>Default:</i> NO | Indicates whether statistics are collected for output queue. |

6.4.3 GUI Configuration

This section describes how to configure RIO in the GUI.

Configuring RIO Parameters

To configure the RIO parameters, perform the following steps:

1. Go to one of the following locations:
 - To set wireless subnet properties, go to **Wireless Subnet Properties Editor > Network Layer > Schedulers and Queues**.
 - To set properties for a specific node, go to **Node Properties Editor > Node Configuration > Network Layer > Schedulers and Queues**.
 - To set properties for a specific interface of a node, go to one of the following locations:
 - **Interface Properties Editor > Interfaces > Interface # > Network Layer > Schedulers and Queues**
 - **Default Device Properties Editor > Interfaces > Interface # > Network Layer > Schedulers and Queues**.

In this section, we show how to configure RIO parameters for a specific node using the Default Device Properties Editor. Parameters can be set in the other properties editors in a similar way.

2. Set **Number of IP Output Queues** to the desired value.

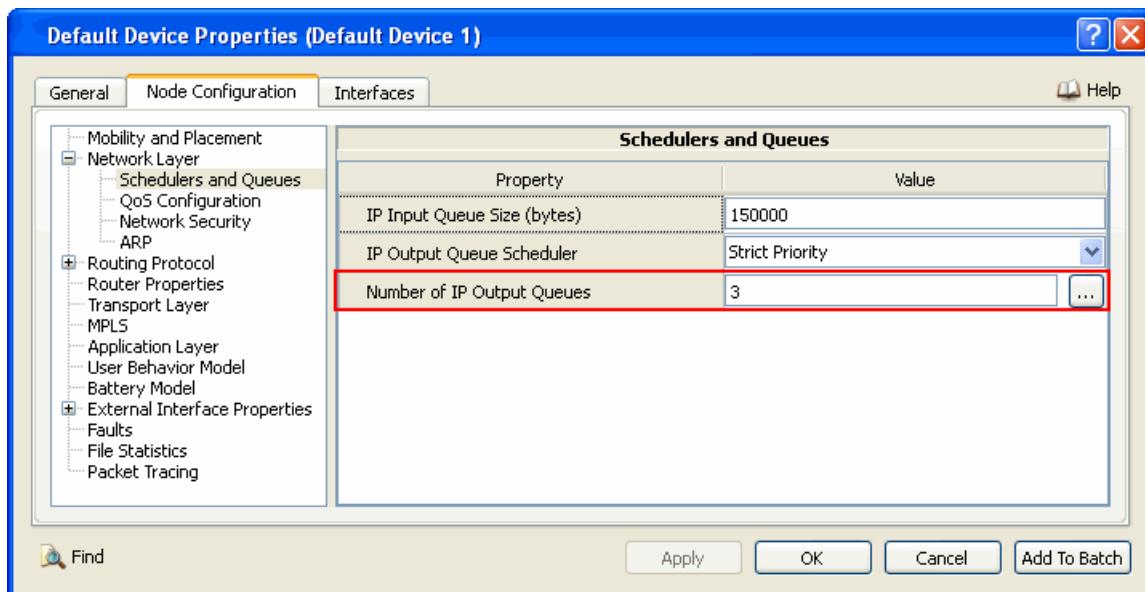


FIGURE 6-7. Setting Number of Queues

3. Click on the Open Array Editor  button in the Value column. This opens the Array Editor.
4. In the left panel of the Array Editor, select the index of the queue to be configured. In the right panel, set IP Queue Type to R/O and set the dependent parameters listed in Table 6-17.

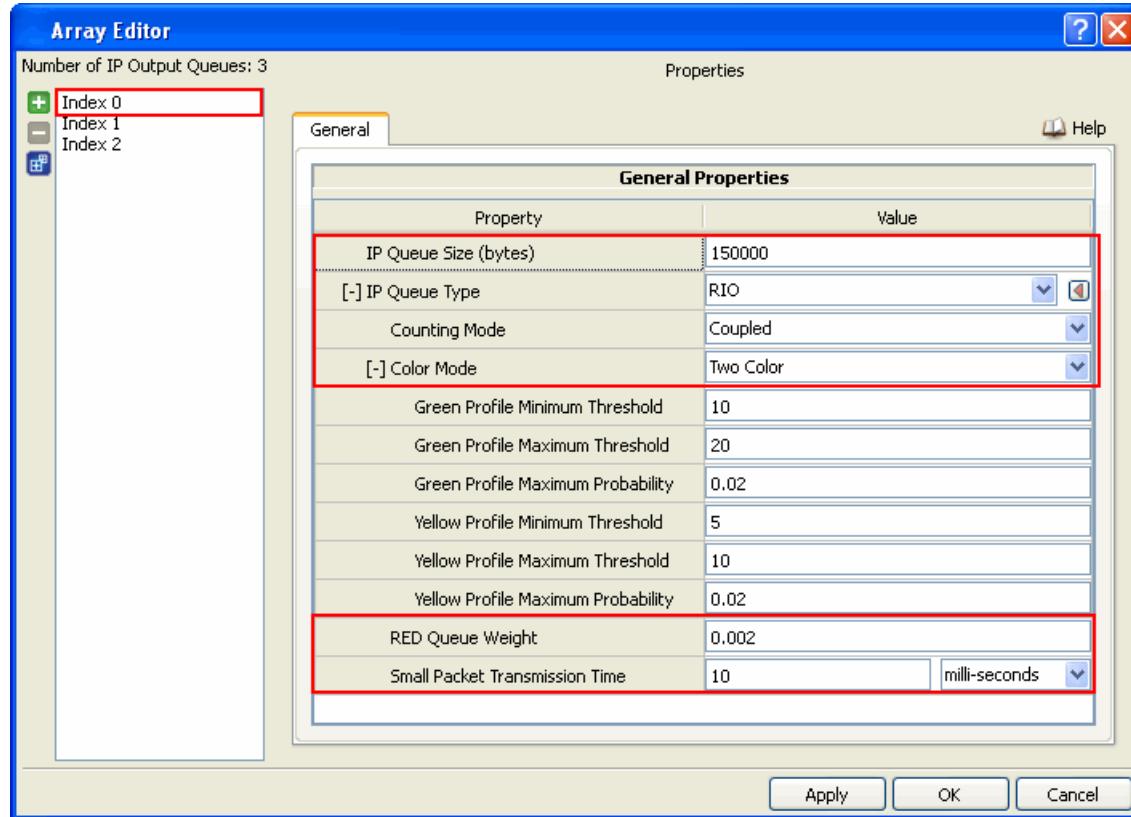


FIGURE 6-8. Setting RIO Queue Parameters

TABLE 6-17. Command Line Equivalent of RIO Queue Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|--------------------------------|-------------------------|------------------------------------|
| IP Queue Size | Node, Interface, Subnet | IP-QUEUE-PRIORITY-QUEUE-SIZE |
| Counting Mode | Node, Interface, Subnet | RIO-COUNTING-MODE |
| Color Mode | Node, Interface, Subnet | RIO-COLOR-MODE |
| RED Queue Weight | Node, Interface, Subnet | RED-QUEUE-WEIGHT |
| Small Packet Transmission Time | Node, Interface, Subnet | RED-SMALL-PACKET-TRANSMISSION-TIME |

5. Set the dependent parameters listed in [Table 6-18](#) for the selected value of **Color Mode**. [Figure 6-9](#) shows the parameters when **Color Mode** is set to *Three Color*. A subset of these parameters are applicable when **Color Mode** is set to *Two Color*.

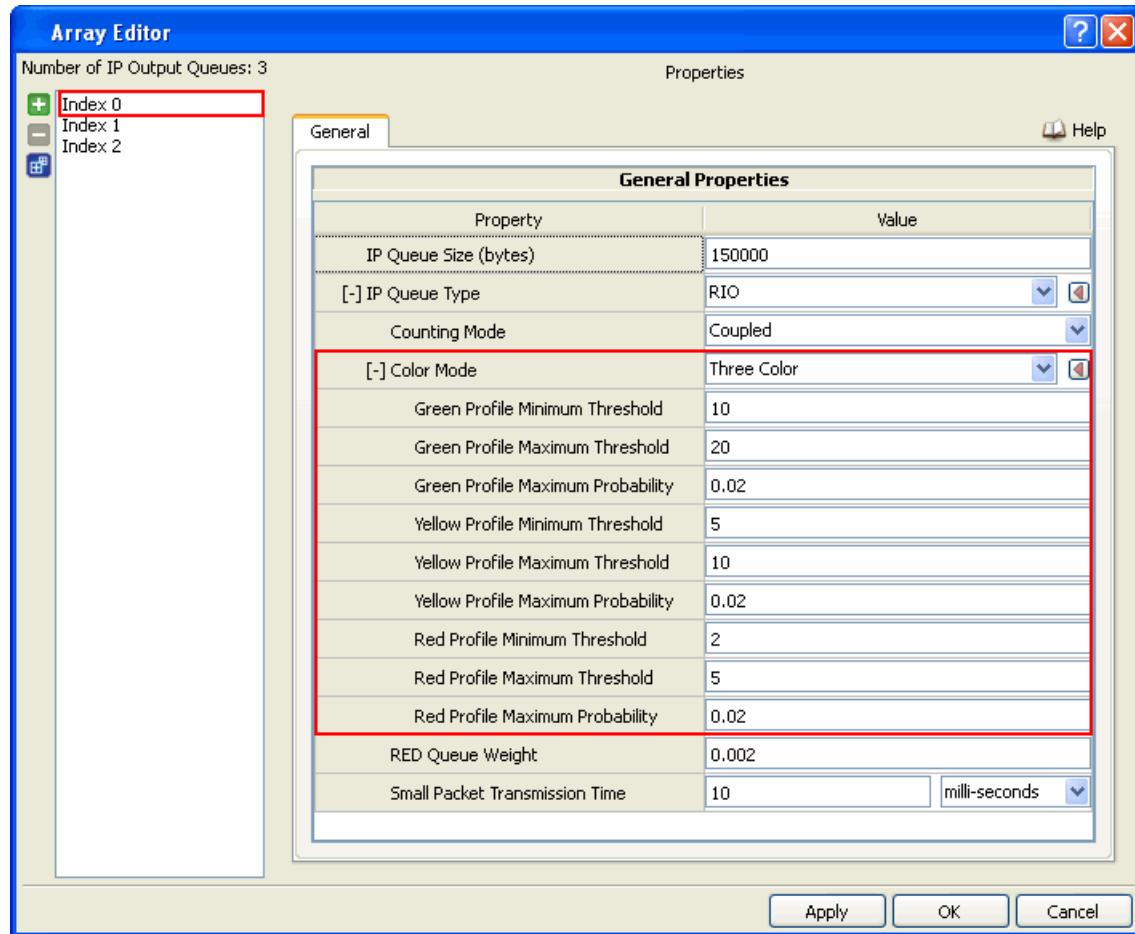


FIGURE 6-9. Setting Color Mode Parameters

TABLE 6-18. Command Line Equivalent of Color Mode Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|------------------------------------|-------------------------|--------------------------------|
| Green Profile Minimum Threshold | Node, Interface, Subnet | GREEN-PROFILE-MIN-THRESHOLD |
| Green Profile Maximum Threshold | Node, Interface, Subnet | GREEN-PROFILE-MAX-THRESHOLD |
| Green Profile Maximum Probability | Node, Interface, Subnet | GREEN-PROFILE-MAX-PROBABILITY |
| Yellow Profile Minimum Threshold | Node, Interface, Subnet | YELLOW-PROFILE-MIN-THRESHOLD |
| Yellow Profile Maximum Threshold | Node, Interface, Subnet | YELLOW-PROFILE-MAX-THRESHOLD |
| Yellow Profile Maximum Probability | Node, Interface, Subnet | YELLOW-PROFILE-MAX-PROBABILITY |
| Red Profile Minimum Threshold | Node, Interface, Subnet | RED-PROFILE-MIN-THRESHOLD |
| Red Profile Maximum Threshold | Node, Interface, Subnet | RED-PROFILE-MAX-THRESHOLD |
| Red Profile Maximum Probability | Node, Interface, Subnet | RED-PROFILE-MAX-PROBABILITY |

Configuring Statistics Parameters

Statistics for RIO can be collected at the global, node, subnet, and interface levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for IP input queues, check the box labeled **IP Input Queue** in the appropriate properties editor.

To enable statistics collection for IP output queues, check the box labeled **IP Output Queue** in the appropriate properties editor.

TABLE 6-19. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-----------------|---------------------------------|------------------------|
| IP Input Queue | Global, Node, Subnet, Interface | INPUT-QUEUE-SATISTICS |
| IP Output Queue | Global, Node, Subnet, Interface | QUEUE-SATISTICS |

6.4.4 Statistics

[Table 6-20](#) lists the RIO statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 6-20. RIO Statistics

| Statistic | Definition |
|--------------------------------|--|
| GREEN Packets Queued | Total number of GREEN packets queued in this queue |
| GREEN Packets Dequeued | Total number of GREEN packets dequeued from this queue |
| GREEN Packets Marked ECN | Total number of GREEN packets marked by the Explicit Congestion Notification (ECN) capable network |
| GREEN Packets Dropped | Total number of GREEN packets dropped by this queue |
| YELLOW Packets Queued | Total number of YELLOW packets queued in this queue |
| YELLOW Packets Dequeued | Total number of YELLOW packets dequeued from this queue |
| YELLOW Packets Marked ECN | Total number of YELLOW packets marked by the Explicit Congestion Notification (ECN) capable network |
| YELLOW Packets Dropped | Total number of YELLOW packets dropped by this queue |
| RED Packets Queued | Total number of RED packets queued in this queue |
| RED Packets Dequeued | Total number of RED packets dequeued from this queue |
| RED Packets Marked ECN | Total number of RED packets marked by the Explicit Congestion Notification (ECN) capable network |
| RED Packets Dropped | Total number of RED packets dropped by this queue |
| (<Profile>) Packets Queued | Total number of <profile> packets queued in this queue |
| (<Profile>) Packets Dequeued | Total number of <profile> packets dequeued from this queue |
| (<Profile>) Packets Marked ECN | Total number of <profile> packets marked by the Explicit Congestion Notification (ECN) capable network |
| (<Profile>) Packets Dropped | Total number of <profile> packets dropped by this queue |
| Total Packets Queued | Total number of packets queued in this queue |
| Total Packets Dequeued | Total number of packets dequeued from this queue |

TABLE 6-20. RIO Statistics (Continued)

| Statistic | Definition |
|----------------------------------|---|
| Total Packets Dropped | Total number of packets dropped due to the queue being full |
| Total Packets Dropped Forcefully | Total number of packets dropped forcefully due to reasons other than the queue being full, such as the next hop not being reachable |
| Average Queue Length (bytes) | Average of queue length reached by this queue during simulation (bytes). See note below. |
| Average Time In Queue | Average delay suffered by the packets in this queue (in seconds). |
| Longest Time in Queue | Longest delay suffered by a packet in this queue (in seconds). |
| Peak Queue Size (bytes) | Largest size ever reached by the queue (bytes) |
| Packets Marked ECN | Total number of packets marked by ECN capable network |

Note: The average queue length is calculated as follows:

$$\text{average queue length} = \frac{\text{sum of weighted packet sizes for all packets}}{\text{(simulation time, in seconds)}},$$

where

$$\text{weighted packet size} = (\text{packet size, in bytes}) * (\text{time spent by the packet in the queue, in seconds})$$

6.4.5 Scenarios Included in QualNet

The QualNet distribution includes several sample scenarios for the RIO queue. All scenarios are located in the directory `QUALNET_HOME/scenarios/developer`. [Table 6-21](#) lists the sub-directory where each scenario is located.

TABLE 6-21. RIO Scenarios Included in QualNet

| Scenario | Description |
|-------------------------|---|
| queue/rio | Shows how packets are handled by RIO queue when bottleneck condition arrives. |
| ecn/rio/rio-with-ecn | Shows the behavior of the RIO with ECN. |
| ecn/rio/rio-without-ecn | Shows the behavior of the RIO without ECN |

6.4.6 References

1. Rupinder Makkar and Ioannis Lambadaris, "Empirical Study Of Buffer Management Schemes For DiffServ Assured Forwarding PHB" (www.sce.carleton.ca/faculty/lambadaris/recent-papers/162.pdf)

6.5 Round Robin Scheduler

6.5.1 Description

The Round Robin scheduler is an alternative scheduling discipline in QualNet. It services each priority queue, starting with the highest priority queue that contains packets, services a single packet, and moves to the next lower priority queue that contains packets, servicing a single packet from each, until each queue with packets has been serviced once. It then starts the cycle over with the highest priority queue containing packets.

6.5.2 Command Line Configuration

To select the Round Robin scheduler, include the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] IP-QUEUE-SCHEDULER      ROUND-ROBIN
```

The scope of this parameter declaration can be Global, Node, Subnet, or Interface. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

Round Robin Scheduler Parameters

[Table 6-22](#) shows the Round Robin scheduler parameters. See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 6-22. Round Robin Parameters

| Parameter | Value | Description |
|---|--|---|
| INPUT-SCHEDULER-STATISTICS <i>Optional</i> Scope: All | List: • YES • NO <i>Default: NO</i> | Indicates whether statistics are collected for input schedulers. |
| SCHEDULER-STATISTICS <i>Optional</i> Scope: All | List: • YES • NO <i>Default: NO</i> | Indicates whether statistics are collected for output schedulers. |

6.5.3 GUI Configuration

This section describes how to configure the Round Robin scheduler in the GUI.

Configuring Round Robin Scheduler Parameters

To configure the Round Robin scheduler parameters, perform the following steps:

1. Go to one of the following locations:

- To set wireless subnet properties, go to **Wireless Subnet Properties Editor > Network Layer > Schedulers and Queues**.

- To set properties for a specific node, go to **Node Properties Editor > Node Configuration > Network Layer > Schedulers and Queues**.
- To set properties for a specific interface of a node, go to one of the following locations:
 - **Interface Properties Editor > Interfaces > Interface # > Network Layer > Schedulers and Queues**.
 - **Default Device Properties Editor > Interfaces > Interface # > Network Layer > Schedulers and Queues**.

In this section, we show how to configure round Robin parameters for a specific node using the Default Device Properties Editor. Parameters can be set in the other properties editors in a similar way.

2. Set **IP Output Queue Scheduler** to *Round Robin* as shown in [Figure 6-10](#).

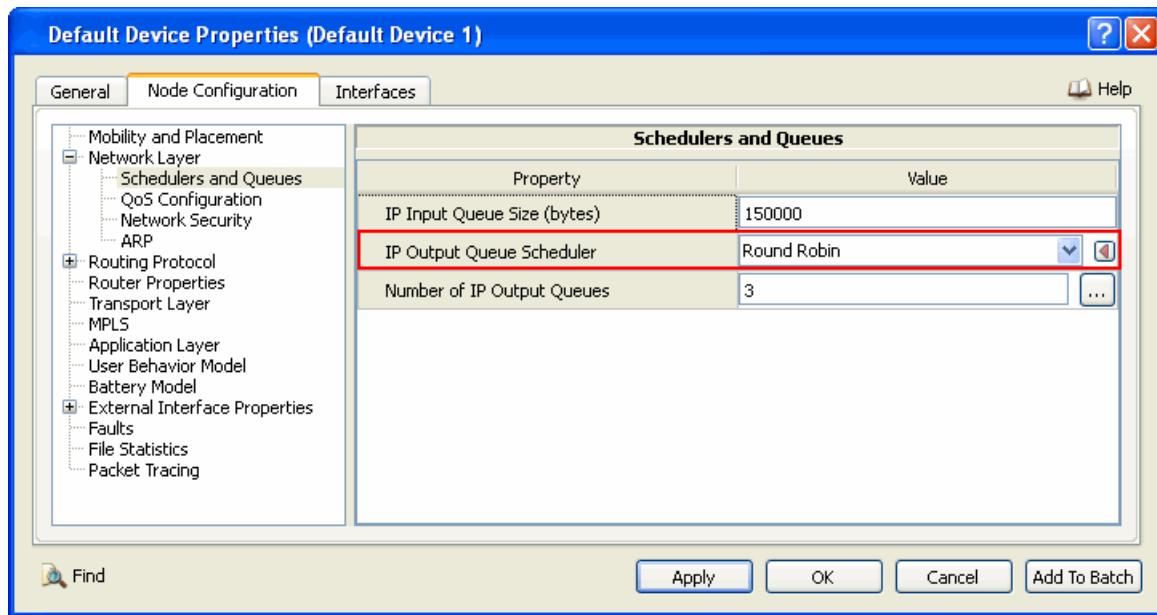


FIGURE 6-10. Setting Round Robin Scheduler Parameters

Configuring Statistics Parameters

Statistics for the Round Robin scheduler can be collected at the global, node, subnet, and interface levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for IP Input Queue Scheduler, check the box labeled **IP Input Scheduler** in the appropriate properties editor.

To enable statistics collection for IP Output Queue Scheduler, check the box labeled **IP Output Scheduler** in the appropriate properties editor.

TABLE 6-23. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------------|---------------------------------|----------------------------|
| IP Input Scheduler | Global, Node, Subnet, Interface | INPUT-SCHEDULER-STATISTICS |
| IP Output Scheduler | Global, Node, Subnet, Interface | SCHEDULER-STATISTICS |

6.5.4 Statistics

[Table 6-24](#) lists the Round Robin statistics that are output to the statistics (.stat) file at the end of simulation:

TABLE 6-24. Round Robin Scheduler Statistics

| Statistics | Description |
|------------------------|--|
| Packets Queued | Total number of packets en queued in the queue. |
| Packets Dequeued | Total number of packets de queued. |
| Packets Dropped | Total number of packets dropped. |
| Service Ratio received | Total number of packets de queued to total de queue requests (see note). |

Note: The service ratio received is calculated as follows:

- If total dequeue requests are not zero, service ratio = (total packets dequeued) / (total dequeue requests)
- If total dequeue requests are equal to zero, service ratio = 0.0.

6.5.5 Scenarios Included in QualNet

The QualNet distribution includes several sample scenarios for the Round Robin scheduler. All scenarios are located in the directory `QUALNET_HOME/scenarios/developer/scheduler`. [Table 6-25](#) lists the sub-directory where each scenario is located.

TABLE 6-25. Round Robin Scenarios Included in QualNet

| Scenario | Description |
|-------------|---|
| round-robin | Shows the behavior of ROUND-ROBIN scheduler at intermediate node. |

6.6 Self-Clocked Fair Queueing (SCFQ) Scheduler

6.6.1 Description

Self-Clocked Fair Queueing (SCFQ) scheduler is similar to Weighted Fair Queueing (WFQ) in its attempt to service each priority queue based on a percentage allocation of the total outgoing bandwidth of the link. However, the calculation of the Finish Time (the time at which a packet would have been serviced given a hypothetical fluid server) for WFQ is complicated. SCFQ uses a simplified method of calculating the service time based on the transmission delay of the packet and the finish time of the packet currently being serviced.

6.6.2 Features and Assumptions

This section describes the implemented features, omitted features, assumptions and limitations of the SFCQ model.

6.6.2.1 Implemented Features

- SCFQ speeds up virtual time computation with reduced complexity by using the finish number of the packet currently in service (CF) as the virtual time function.

6.6.2.2 Omitted Features

None.

6.6.2.3 Assumptions and Limitations

- Users can specify a weight for each queue but it should be <1. If there is no weight specification for queues, weights for each queue are assigned internally by their priority.

6.6.3 Command Line Configuration

To select the SCFQ scheduler, include the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] IP-QUEUE-SCHEDULER SELF-CLOCKED-FAIR
```

The scope of this parameter declaration can be Global, Node, Subnet, or Interface. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

SCFQ Parameters

SCFQ scheduler parameters are described in [Table 6-26](#). See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 6-26. SCFQ Parameters

| Parameter | Value | Description |
|--|---|---|
| QUEUE-WEIGHT <i>Optional</i> Scope: All Instances: priority | Real <i>Range:</i> (0.0 , 1.0] <i>Default:</i> See note | Weight of the queue. Note: The weights for all queues on an interface should add up to 1. |
| INPUT-SCHEDULER-STATISTICS <i>Optional</i> Scope: All | List • YES • NO <i>Default:</i> NO | Indicates whether statistics are collected for input schedulers. |
| SCHEDULER-STATISTICS <i>Optional</i> Scope: All | List • YES • NO <i>Default:</i> NO | Indicates whether statistics are collected for output schedulers. |

Note: The default weight of each queue is the ((priority value of the queue)+ 1) / (sum of all the priority values for all the queues + the number of queues).

6.6.4 GUI Configuration

This section describes how to configure Self Clocked Fair Queueing Scheduler in the GUI.

Configuring SCFQ Scheduler Parameters

To configure the Self-Clocked Fair Queueing (SCFQ) Scheduler parameters, perform the following steps:

1. Go to one of the following locations:
 - To set wireless subnet properties, go to **Wireless Subnet Properties Editor > Network Layer > Schedulers and Queues**.
 - To set properties for a specific node, go to **Node Properties Editor > Node Configuration > Network Layer > Schedulers and Queues**.
 - To set properties for a specific interface of a node, go to one of the following locations:
 - **Interface Properties Editor > Interfaces > Interface # > Network Layer > Schedulers and Queues**
 - **Default Device Properties Editor > Interfaces > Interface # > Network Layer > Schedulers and Queues**

In this section, we show how to configure SCFQ parameters for a specific node using the Default Device Properties Editor. Parameters can be set in the other properties editors in a similar way.

2. Set IP Output Queue Scheduler to Self Clocked Fair.

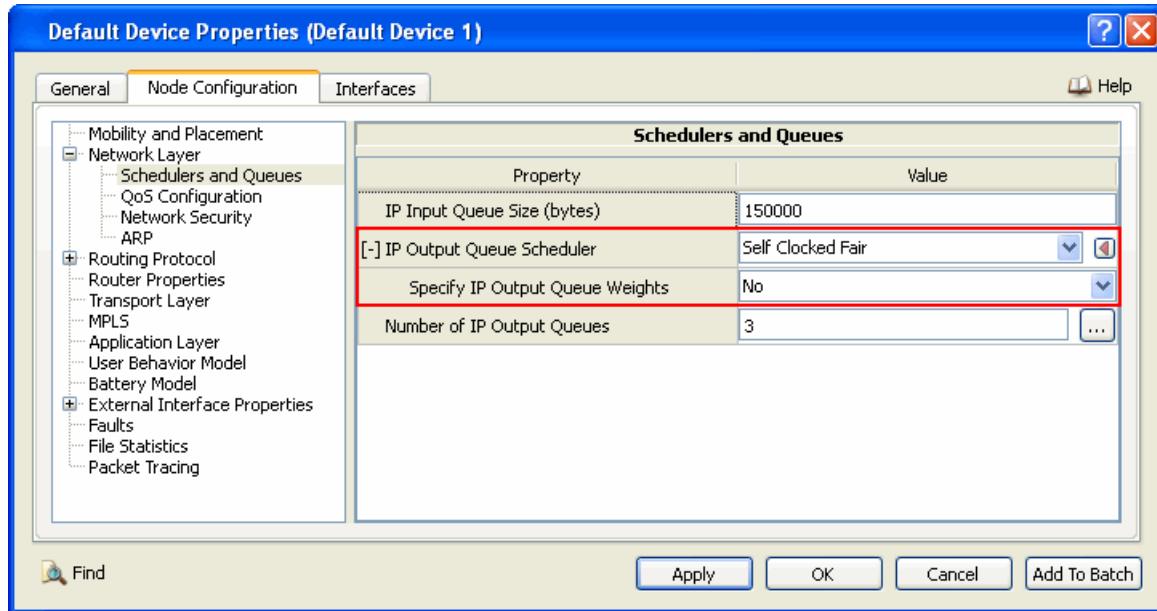


FIGURE 6-11. Setting SCFQ Parameters

Setting Parameters

- To specify IP Queue Weights, set **Specify Priority Output Queue Weights** to Yes; otherwise, set **Specify IP Output Queue Weights** to No.
- If **Specify IP Output Queue Weights** is set to Yes, then configure the queue weights as follows:
 - Set **Number of IP Output Queues** to the desired value.
 - Click on the **Open Array Editor** button in the **Value** column. This opens the Array Editor.
 - In the left panel of the Array Editor, select the index of the queue to be configured. In the right panel, set the queue weight parameters listed in [Table 6-27](#) and the other parameters for the queue.

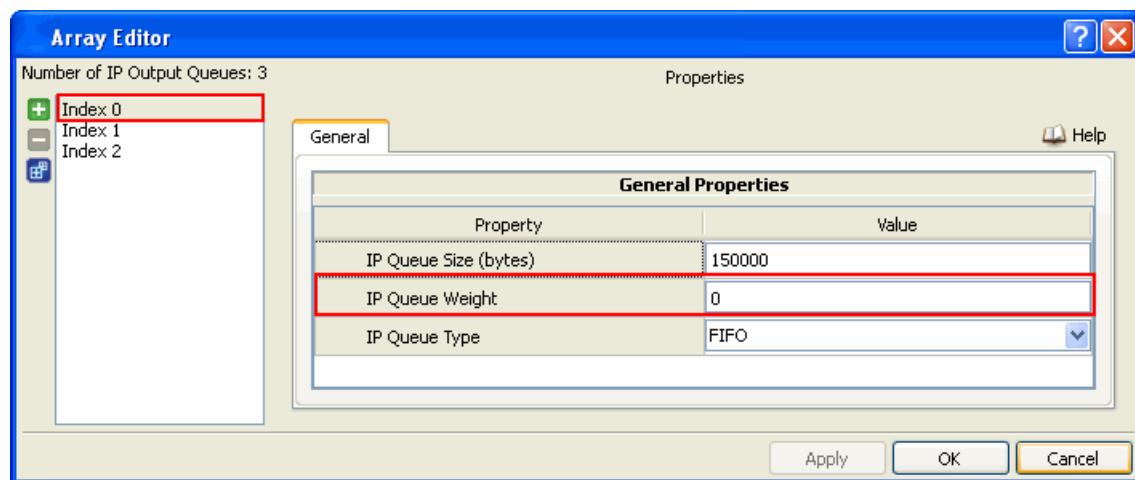


FIGURE 6-12. Setting Queue Weight Parameters

TABLE 6-27. Command Line Equivalent of Queue Weight Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-----------------|-------------------------|------------------------|
| IP Queue Weight | Node, Subnet, Interface | QUEUE-WEIGHT |

Configuring Statistics Parameters

Statistics for the Self-Clocked Fair Queueing (SCFQ) scheduler can be collected at the global, node, subnet, and interface levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for IP Input Queue Scheduler, check the box labeled **IP Input Scheduler** in the appropriate properties editor.

To enable statistics collection for IP Output Queue Scheduler, check the box labeled **IP Output Scheduler** in the appropriate properties editor.

TABLE 6-28. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------------|---------------------------------|----------------------------|
| IP Input Scheduler | Global, Node, Subnet, Interface | INPUT-SCHEDULER-STATISTICS |
| IP Output Scheduler | Global, Node, Subnet, Interface | SCHEDULER-STATISTICS |

6.6.5 Statistics

[Table 6-29](#) lists the SCFQ statistics that are output to the statistics (.stat) file at the end of simulation:

TABLE 6-29. SCFQ Statistics

| Statistic | Description |
|-------------------------------------|---|
| Packets Queued | Total number of packets enqueued in the queue |
| Packets Dequeued | Total number of packets dequeued |
| Packets Dropped | Total number of packets dropped |
| Dequeue Requests While Queue Active | Total number of dequeue requests while the queue is active. |
| Dequeue Requests Serviced | Total number of de queue requests serviced. See note 1 below. |
| Service Ratio Received | Total number of serviced dequeue request to total dequeue requests. See note 2 below. |
| Total Service Received(Bytes) | Total service received in bytes. |

Notes: Dequeue requests serviced is calculated as follows:

1. Dequeue requests serviced = total number of dequeue requests - total number of missed services.
2. Service ratio received is calculated as follows:
 - If total dequeue requests are greater than zero, service ratio received = (total number of missed services) / (total number of dequeue requests)
 - If total dequeue requests is equal to zero, service ratio received = 0.0

6.6.6 Scenarios Included in QualNet

The QualNet distribution includes several sample scenarios for the SCFQ scheduler. All scenarios are located in the directory `QUALNET_HOME/scenarios/developer/scheduler`. [Table 6-30](#) lists the sub-directory where each scenario is located.

TABLE 6-30. SCFQ Scenarios Included in QualNet

| Scenario | Description |
|----------|--|
| scfq | Shows the behavior of SCFQ scheduler at the intermediate node. |

6.6.7 References

The QualNet SCFQ model is based on the information available at the following URLs:

1. <http://www.cisco.com/warp/public/732/Tech/wfq/>
2. www.rennes.enst-bretagne.fr/~toutain/G6Recherche/Papier-LBSCFQ-lamti.PDF

6.7 Strict Priority Scheduler

6.7.1 Description

The Strict Priority scheduler services the highest priority queue until it is empty, and then moves to the next highest priority queue, and so on. It is possible that if there is enough high priority traffic, the lower priorities could be starved.

6.7.2 Command Line Configuration

To select the Strict Priority scheduler, include the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] IP-QUEUE-SCHEDULER      STRICT-PRIORITY
```

The scope of this parameter declaration can be Global, Node, Subnet, or Interface. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

Strict Priority Scheduler Parameters

[Table 6-31](#) shows the Strict Priority scheduler parameters. See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 6-31. Strict Priority Scheduler Parameters

| Parameter | Value | Description |
|---|---|---|
| INPUT-SCHEDULER-STATISTICS <i>Optional</i> Scope: All | List • YES • NO <i>Default: NO</i> | Indicates whether statistics are collected for input schedulers. |
| SCHEDULER-STATISTICS <i>Optional</i> Scope: All | List • YES • NO <i>Default: NO</i> | Indicates whether statistics are collected for output schedulers. |

6.7.3 GUI Configuration

To configure the Strict Priority Scheduler, perform the following steps:

1. Go to one of the following locations:
 - To set properties for a specific wireless subnet, go to **Wireless Subnet Properties Editor > Network Layer > Schedulers and Queues**.
 - To set properties for a specific node, go to **Default Device Properties Editor > Node Configuration > Network Layer > Schedulers and Queues**.

- To set properties for a specific interface of a node, go to one of the following locations:
 - Interface Properties Editor > Interfaces > Interface # > Network Layer > Schedulers and Queues.**
 - Default Device Properties Editor > Interfaces > Interface # > Network Layer > Schedulers and Queues.**

In this section, we show how to configure the Strict Priority Scheduler parameters for a specific node using the Default Device Properties Editor. Parameters can be set in the other properties editors in a similar way.

- Set **IP Output Queue Scheduler** to *Strict Priority*. There are no dependent parameters for this scheduler.

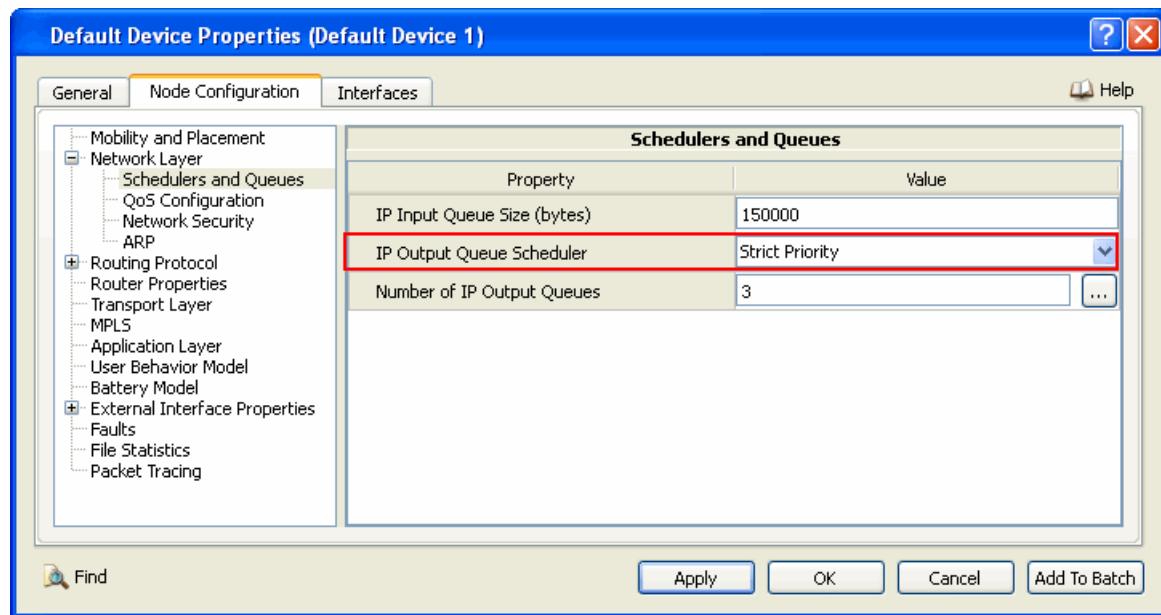


FIGURE 6-13. Setting Strict Priority Scheduler

Configuring Statistics Parameters

Statistics for the Strict Priority scheduler can be collected at the global, node, subnet, and interface levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for IP Input Queue Scheduler, check the box labeled **IP Input Scheduler** in the appropriate properties editor.

To enable statistics collection for IP Output Queue Scheduler, check the box labeled **IP Output Scheduler** in the appropriate properties editor.

TABLE 6-32. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------------|---------------------------------|----------------------------|
| IP Input Scheduler | Global, Node, Subnet, Interface | INPUT-SCHEDULER-STATISTICS |
| IP Output Scheduler | Global, Node, Subnet, Interface | SCHEDULER-STATISTICS |

6.7.4 Statistics

[Table 6-33](#) lists the Strict Priority scheduler statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 6-33. Strict Priority Scheduler Statistics

| Statistics | Description |
|------------------|---|
| Packets Queued | Total number of packets enqueued in the queue |
| Packets Dequeued | Total number of packets dequeued |
| Packets Dropped | Total number of packets dropped |

6.7.5 Scenarios Included in QualNet

The QualNet distribution includes several sample scenarios for the Strict Priority scheduler. All scenarios are located in the directory `QUALNET_HOME/scenarios/developer/scheduler`. [Table 6-34](#) lists the sub-directory where each scenario is located.

TABLE 6-34. Strict Priority Scheduler Scenarios Included in QualNet

| Scenario | Description |
|-----------------|---|
| strict-priority | Shows the behavior of STRICT-PRIORITY scheduler at the intermediate node. Packets with priority 0 are considered to be the highest priority and will be served as soon as they arrive at the intermediate node, leading to the phenomenon of lower priority queue starvation. |

6.8 Weighted Fair Queuing (WFQ) Scheduler

6.8.1 Description

WFQ Scheduler is an alternative scheduling discipline in QualNet. WFQ attempts to service each priority queue based on a percentage allocation of the total outgoing bandwidth of the link. This allocation is based on a configurable weight assigned to each queue.

6.8.2 Features and Assumptions

This section describes the implemented features, omitted features, assumptions and limitations of the WFQ scheduler model.

6.8.2.1 Implemented Features

- Selects the queue with the smallest virtual finish time (SF) for dequeue.

6.8.2.2 Omitted Features

None.

6.8.2.3 Assumptions and Limitations

- Users can specify a weight for each queue, but it should be >0 and <1. If there is no weight specified for the queues, weights for each queue are assigned internally according to their priority.

6.8.3 Command Line Configuration

To select the WFQ scheduler, include the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] IP-QUEUE-SCHEDULER WEIGHTED-FAIR
```

The scope of this parameter declaration can be Global, Node, Subnet, or Interface. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

WFQ Parameters

WFQ parameters are described in [Table 6-35](#). See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 6-35. SCFQ Parameters

| Parameter | Value | Description |
|--|--|---|
| QUEUE-WEIGHT <i>Optional</i> Scope: All Instances: priority | Real <i>Range:</i> [0.0 , 1.0] <i>Default:</i> See note below table. | Weight of the queue. Note: The weights for all queues on an interface should add up to 1. |
| INPUT-SCHEDULER-STATISTICS <i>Optional</i> Scope: All | List • YES • NO <i>Default:</i> NO | Indicates whether statistics are collected for input schedulers. |
| SCHEDULER-STATISTICS <i>Optional</i> Scope: All | List • YES • NO <i>Default:</i> NO | Indicates whether statistics are collected for output schedulers. |

Note: The default weight of each queue is the ((priority value of the queue)+ 1) / (sum of all the priority values for all the queues + number of queues).

6.8.4 GUI Configuration

This section describes how to configure Weighted Fair Queuing (WFQ) Scheduler model in the GUI.

Configuring WFQ Parameters

To configure the Weighted Fair Queuing Scheduler parameters, perform the following steps:

1. Go to one of the following locations:
 - To set wireless subnet properties, go to **Wireless Subnet Properties Editor > Network Layer > Schedulers and Queues**.
 - To set properties for a specific node, go to **Node Properties Editor > Node Configuration > Network Layer > Schedulers and Queues**.
 - To set properties for a specific interface of a node, go to one of the following locations:
 - **Interface Properties Editor > Interfaces > Interface # > Network Layer > Schedulers and Queues**.
 - **Default Device Properties Editor > Interfaces > Interface # > Network Layer > Schedulers and Queues**.

In this section, we show how to configure WFQ parameters for a specific node using the Default Device Properties Editor. Parameters can be set in the other properties editors in a similar way.

2. Set IP Output Queue Scheduler to *Weighted Fair*.

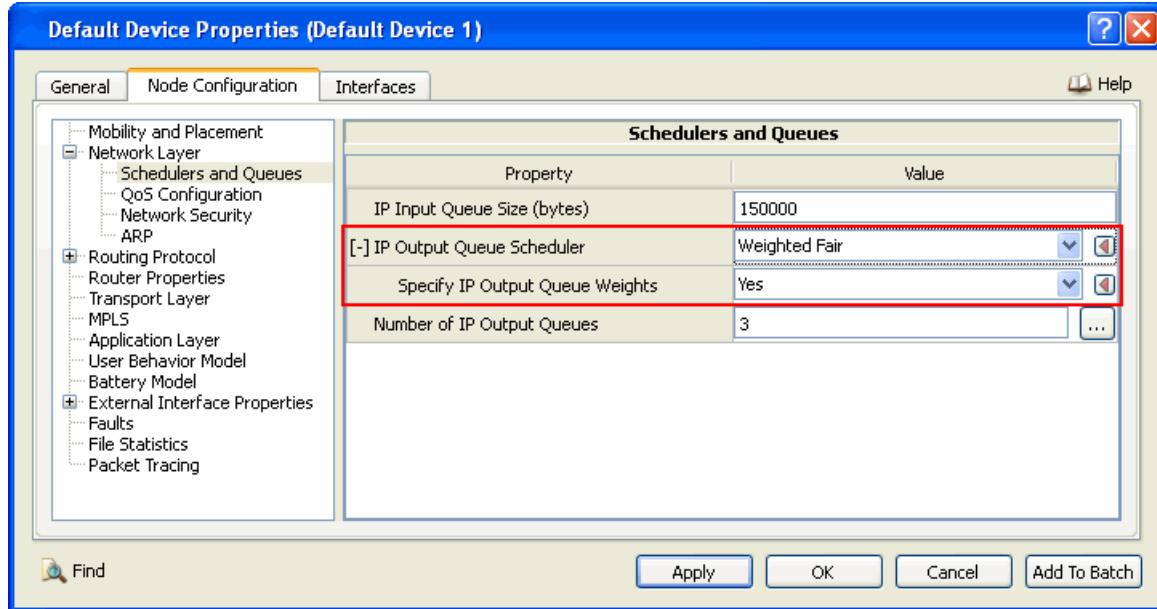


FIGURE 6-14. Setting WFQ Parameters

Setting Parameters

- To specify IP Queue Weights, set **Specify IP Output Queue Weights** to Yes; otherwise, set **Specify IP Output Queue Weights** to No.
3. If **Specify IP Output Queue Weights** is set to Yes, then configure the queue weights as follows:
- Set **Number of IP Output Queues** to the desired value.
 - Click on the **Open Array Editor** button in the **Value** column. This opens the Array Editor.
 - In the left panel of the Array Editor, select the index of the queue to be configured. In the right panel, set the queue weight parameters listed in [Table 6-36](#) and the other parameters for the queue.

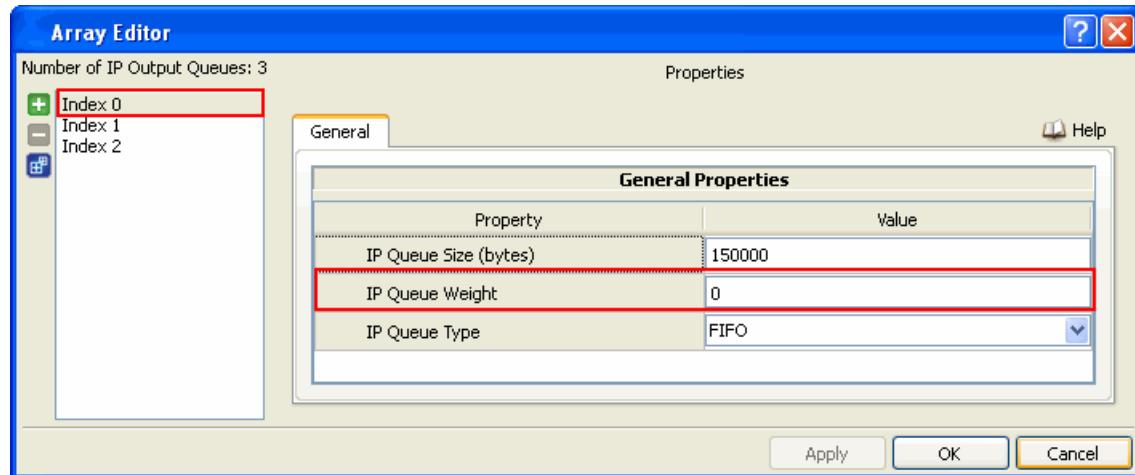


FIGURE 6-15. Setting Queue Weight Parameters

TABLE 6-36. Command Line Equivalent of Queue Weight Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-----------------|-------------------------|------------------------|
| IP Queue Weight | Node, Subnet, Interface | QUEUE-WEIGHT |

Configuring Statistics Parameters

Statistics for the Weighted Fair Queue scheduler can be collected at the global, node, subnet, and interface levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for IP Input Queue Scheduler, check the box labeled **IP Input Scheduler** in the appropriate properties editor.

To enable statistics collection for IP Output Queue Scheduler, check the box labeled **IP Output Scheduler** in the appropriate properties editor.

TABLE 6-37. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------------|---------------------------------|----------------------------|
| IP Input Scheduler | Global, Node, Subnet, Interface | INPUT-SCHEDULER-STATISTICS |
| IP Output Scheduler | Global, Node, Subnet, Interface | SCHEDULER-STATISTICS |

6.8.5 Statistics

[Table 6-38](#) lists the WFQ scheduler statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 6-38. WFQ Scheduler Statistics

| Statistic | Description |
|-------------------------------------|---|
| Packets Queued | Total number of packets enqueued in the queue. |
| Packets Dequeued | Total number of packets dequeued. |
| Packets Dropped | Total number of packets dropped. |
| Dequeue Requests While Queue Active | Total number of dequeue requests while the queue is active. |
| Dequeue Requests Serviced | Total number of dequeue requests services. |
| Service Ratio received | Total packets dequeued to total dequeue requests. |
| Total Service Received(Bytes) | Total service received in bytes. |

6.8.6 Scenarios Included in QualNet

The QualNet distribution includes several sample scenarios for the WFQ scheduler model. All scenarios are located in the directory `QUALNET_HOME/scenarios/developer/scheduler`. [Table 6-39](#) lists the sub-directory where each scenario is located.

TABLE 6-39. WFQ Scheduler Scenarios Included in QualNet

| Scenario | Description |
|----------|---|
| wfq | Shows the behavior of WFQ scheduler at the intermediate node. |

6.8.7 References

The QualNet WFQ model is based on the information available at the following URLs:

1. Computer Networking by Jame F. Kurose, Keith W. Ross [page - 530].
2. <http://www.cs.berkeley.edu/~kfall/EE122/wfq-notes/index.htm>.
3. Wfq from Cisco <http://www.cisco.com/warp/public/732/Tech/wfq/>.
4. Wfq implementation http://www.sics.se/~ianm/WFQ/wfq_descrip/node21.html.

6.9 Weighted RED (WRED) Queue

6.9.1 Description

WRED is a multilevel RED (MRED) queuing algorithm suitable for the AF PHB that can identify any preferences and can punish misbehaving users. The incoming packets are marked through packet marking algorithm on the basis of Service Level Agreement (SLA) between the customer and service-provider. The marking is done in two-color (DP0 and DP1) or three-color (DP0, DP1, DP2) level referring drop precedence (DP) as below:

- Green: Level Low Drop, DP0
- Yellow: Level Medium Drop, DP1
- Red: Level High Drop, DP2

WRED is a Single Average Multiple threshold (SAMT) variant of multilevel RED and can operate in three-color mode. It uses three RED algorithms - destined for DP0, DP1 and DP2 packets. When congestion occurs it first drops DP2 packets, and if the congestion persists, it drops DP1 packets and finally it drops DP0 packets as a last resort. But how and when it starts dropping DP1 or DP0 packets depends on the RED parameter specification for each color. Thus, RED parameter settings in configuration file should reflect this inherent assumption that it drops DP2 packets more aggressively than dropping the DP1 packets. Again it drops DP1 packets more aggressively than dropping the DP0 packets.

6.9.2 Command Line Configuration

To specify WRED as the queue discipline, include the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] IP-QUEUE-TYPE [<Index>] WRED
```

where

<Index> Queue index to which this parameter declaration is applicable, enclosed in square brackets. This should be in the range 0 to $n-1$, where n is the number of priority queues at the interface.

The instance specification is optional. If an instance is not included, then the parameter declaration is applicable to all queues at the interface

The scope of this parameter declaration can be Global, Node, Subnet, or Interface. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

WRED Parameters

Table 6-40 shows the WRED parameters. See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 6-40. WRED Parameters

| Parameter | Value | Description |
|--|--|--|
| IP-QUEUE-PRIORITY-QUEUE-SIZE <i>Optional</i> Scope: All Instances: queue index | Integer <i>Range:</i> > 0 <i>Default:</i> 150000 <i>Unit:</i> bytes | Queue size. |
| GREEN-PROFILE-MIN-THRESHOLD <i>Optional</i> Scope: All Instances: queue index | Integer <i>Range:</i> > 0 <i>Default:</i> 10 <i>Unit:</i> packets | Specifies the number of green profile packets in the queue that represents the lower bound at which green packets can be randomly dropped. |
| GREEN-PROFILE-MAX-THRESHOLD <i>Optional</i> Scope: All Instances: queue index | Integer <i>Range:</i> > 0 <i>Default:</i> 20 <i>Unit:</i> packets | Specifies the number of green packets in the queue that represents the upper bound at which green packets can be randomly dropped. |
| GREEN-PROFILE-MAX-PROBABILITY <i>Optional</i> Scope: All Instances: queue index | Real <i>Range:</i> [0.0, 1.0] <i>Default:</i> 0.02 | Specifies the maximum probability value at which a packet can be dropped (before the queue is completely full). |
| YELLOW-PROFILE-MIN-THRESHOLD <i>Optional</i> Scope: All Instances: queue index | Integer <i>Range:</i> > 0 <i>Default:</i> 5 <i>Unit:</i> packets | Specifies the number of yellow profile packets in the queue that represents the lower bound at which yellow packets can be randomly dropped. |
| YELLOW-PROFILE-MAX-THRESHOLD <i>Optional</i> Scope: All Instances: queue index | Integer <i>Range:</i> > 0 <i>Default:</i> 10 <i>Unit:</i> packets | Specifies the number of yellow profile packets in the queue that represents the upper bound at which yellow packets can be randomly dropped. |

TABLE 6-40. WRED Parameters (Continued)

| Parameter | Value | Description |
|---|---|--|
| YELLOW-PROFILE-MAX-PROBABILITY <i>Optional</i> Scope: All Instances: queue index | Real <i>Range:</i> [0.0, 1.0] <i>Default:</i> 0.02 | Specifies the maximum probability (0...1) value at which a packet can be dropped (before the queue is completely full). |
| RED-PROFILE-MIN-THRESHOLD <i>Optional</i> Scope: All Instances: queue index | Integer <i>Range:</i> > 0 <i>Default:</i> 2 <i>Unit:</i> packets | Specifies the number of red profile packets in the queue that represents the lower bound at which red packets can be randomly dropped. |
| RED-PROFILE-MAX-THRESHOLD <i>Optional</i> Scope: All Instances: queue index | Integer <i>Range:</i> > 0 <i>Default:</i> 5 <i>Unit:</i> packets | Specifies the number of red profile packets in the queue that represents the upper bound at which red packets can be randomly dropped. |
| RED-PROFILE-MAX-PROBABILITY <i>Optional</i> Scope: All Instances: queue index | Real <i>Range:</i> [0.0, 1.0] <i>Default:</i> 0.02 | Specifies the maximum probability (0...1) value at which a packet can be dropped (before the queue is completely full). |
| RED-QUEUE-WEIGHT <i>Optional</i> Scope: All Instances: queue index | Real <i>Range:</i> [0.0, 1.0] <i>Default:</i> 0.002 | Specifies the queue weight used to determine the bias towards recent or historical queue lengths in calculating the average. |
| RED-SMALL-PACKET-TRANSMISSION-TIME <i>Optional</i> Scope: All Instances: queue index | Time <i>Range:</i> > 0S <i>Default:</i> 10MS | Specifies the sample amount of time to transmit a small packet. Used to estimate the queue average during idle periods. |

TABLE 6-40. WRED Parameters (Continued)

| Parameter | Value | Description |
|---|---|--|
| INPUT-QUEUE-STATISTICS <i>Optional</i> Scope: All | List • YES • NO <i>Default:</i> NO | Indicates whether statistics are collected for input queue. |
| QUEUE-STATISTICS <i>Optional</i> Scope: All | List • YES • NO <i>Default:</i> NO | Indicates whether statistics are collected for output queue. |

6.9.3 GUI Configuration

This section describes how to configure the Weighted RED Queue in the GUI.

Configuring Weighted RED (WRED) Queue Parameters

To configure the Weighted RED (WRED) Queue parameters, perform the following steps:

1. Go to one of the following locations:
 - To set wireless subnet properties, go to **Wireless Subnet Properties Editor > Network Layer > Schedulers and Queues**.
 - To set properties for a specific node, go to **Node Properties Editor > Node Configuration > Network Layer > Schedulers and Queues**.
 - To set properties for a specific interface of a node, go to one of the following locations:
 - **Interface Properties Editor > Interfaces > Interface # > Network Layer > Schedulers and Queues**
 - **Default Device Properties Editor > Interfaces > Interface # > Network Layer > Schedulers and Queues**.

In this section, we show how to configure WRED parameters for a specific node using the Default Device Properties Editor. Parameters can be set in the other properties editors in a similar way.

2. Set **Number of IP Output Queues** to the desired value.

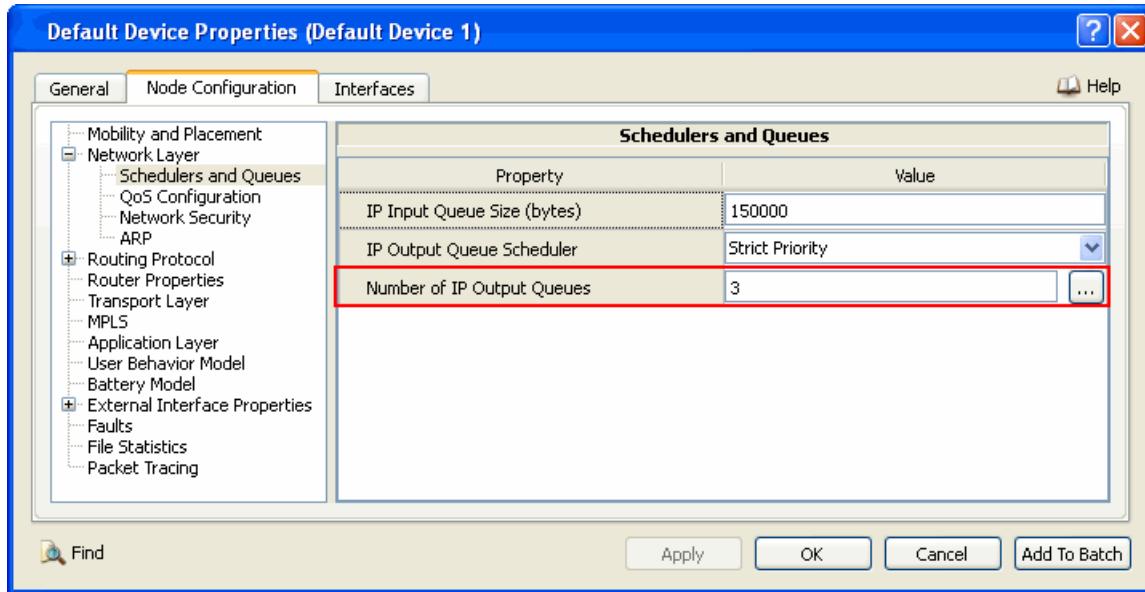


FIGURE 6-16. Setting Number of Queues

3. Click on the Open Array Editor  button in the Value column. This opens the Array Editor.
4. In the left panel of the Array Editor, select the index of the queue to be configured. In the right panel, set IP Queue Type to WRED and set the dependent parameters listed in Table 6-41.

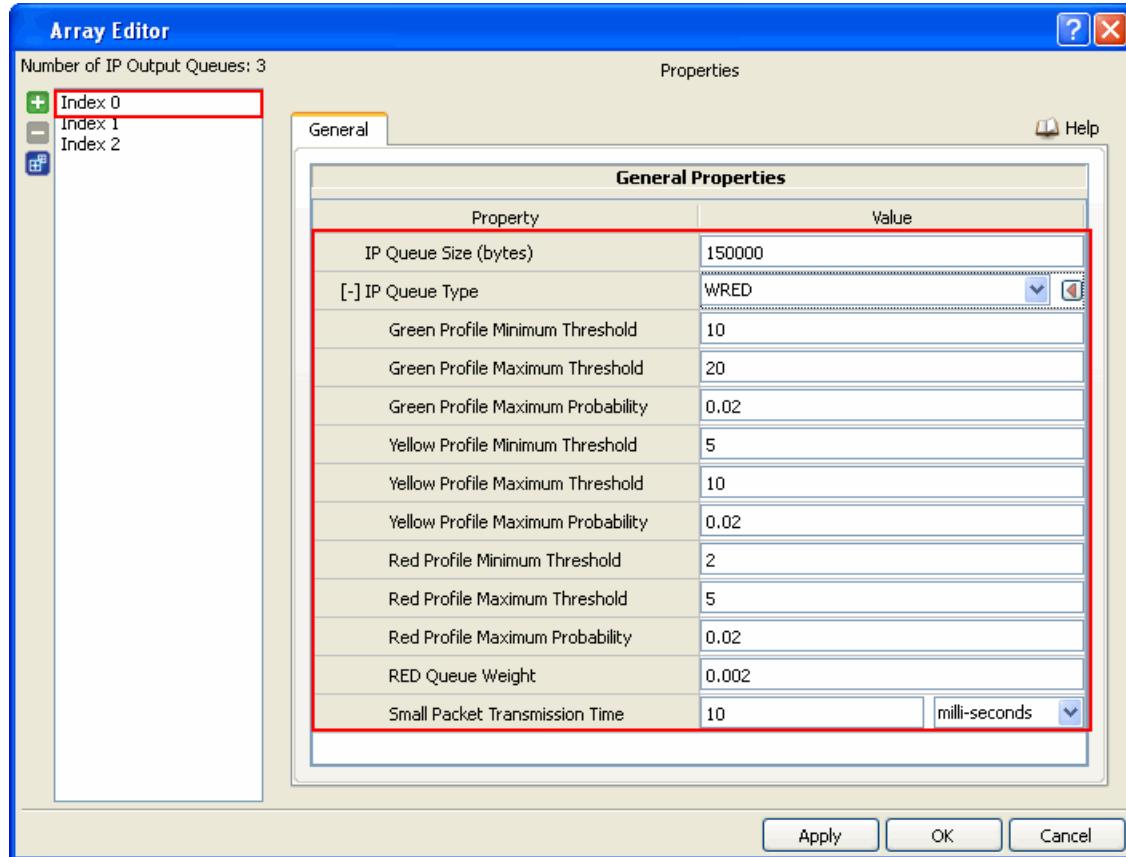


FIGURE 6-17. Setting WRED Queue Parameters

TABLE 6-41. Command Line Equivalent of WRED Queue Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|------------------------------------|-------------------------|--------------------------------|
| IP Queue Size | Node, Subnet, Interface | IP-QUEUE-PRIORITY-QUEUE-SIZE |
| Green Profile Minimum Threshold | Node, Subnet, Interface | GREEN-PROFILE-MIN-THRESHOLD |
| Green Profile Maximum Threshold | Node, Subnet, Interface | GREEN-PROFILE-MAX-THRESHOLD |
| Green Profile Maximum Probability | Node, Subnet, Interface | GREEN-PROFILE-MAX-PROBABILITY |
| Yellow Profile Minimum Threshold | Node, Subnet, Interface | YELLOW-PROFILE-MIN-THRESHOLD |
| Yellow Profile Maximum Threshold | Node, Subnet, Interface | YELLOW-PROFILE-MAX-THRESHOLD |
| Yellow Profile Maximum Probability | Node, Subnet, Interface | YELLOW-PROFILE-MAX-PROBABILITY |
| Red Profile Minimum Threshold | Node, Subnet, Interface | RED-PROFILE-MIN-THRESHOLD |
| Red Profile Maximum Threshold | Node, Subnet, Interface | RED-PROFILE-MAX-THRESHOLD |

TABLE 6-41. Command Line Equivalent of WRED Queue Parameters (Continued)

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------------------------|-------------------------|------------------------------------|
| Red Profile Maximum Probability | Node, Subnet, Interface | RED-PROFILE-MAX-PROBABILITY |
| Red Queue Weight | Node, Subnet, Interface | RED-QUEUE-WEIGHT |
| Small Packet Transmission Time | Node, Subnet, Interface | RED-SMALL-PACKET-TRANSMISSION-TIME |

Configuring Statistics Parameters

Statistics for WRED can be collected at the global, node, subnet, and interface levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for IP input queues, check the box labeled **IP Input Queue** in the appropriate properties editor.

To enable statistics collection for IP output queues, check the box labeled **IP Output Queue** in the appropriate properties editor.

TABLE 6-42. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-----------------|---------------------------------|------------------------|
| IP Input Queue | Global, Node, Subnet, Interface | INPUT-QUEUE-SATISTICS |
| IP Output Queue | Global, Node, Subnet, Interface | QUEUE-SATISTICS |

6.9.4 Statistics

Table 6-43 lists the WRED queue statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 6-43. WRED Statistics

| Statistics | Description |
|---------------------------|--|
| GREEN Packets Queued | Total number of GREEN packets queued in this queue. |
| GREEN Packets Dequeued | Total number of GREEN packets dequeued from this queue. |
| GREEN Packets Marked ECN | Total number of GREEN packets marked by the Explicit Congestion Notification (ECN) capable network. |
| GREEN Packets Dropped | Total number of GREEN packets dropped by this queue. |
| YELLOW Packets Queued | Total number of YELLOW packets queued in this queue. |
| YELLOW Packets Dequeued | Total number of YELLOW packets dequeued from this queue. |
| YELLOW Packets Marked ECN | Total number of YELLOW packets marked by the Explicit Congestion Notification (ECN) capable network. |
| YELLOW Packets Dropped | Total number of YELLOW packets dropped by this queue. |
| RED Packets Queued | Total number of RED packets queued in this queue. |
| RED Packets Dequeued | Total number of RED packets dequeued from this queue. |

TABLE 6-43. WRED Statistics

| Statistics | Description |
|----------------------------------|--|
| RED Packets Marked ECN | Total number of RED packets marked by the Explicit Congestion Notification (ECN) capable network. |
| RED Packets Dropped | Total number of RED packets dropped by this queue. |
| Total Packets Queued | Total number of packets queued in this queue. |
| Total Packets Dequeued | Total number of packets dequeued from this queue. |
| Total Packets Dropped | Total number of packets dropped by this queue. |
| Average Queue Length (bytes) | Average queue length in bytes reached by this queue during simulation. See note below. |
| Average Time in Queue (seconds) | Average delay in seconds suffered by the packets in this queue. |
| Longest Time in Queue (seconds) | Longest delay in seconds suffered by a packet in this queue. |
| Peak Queue Size (bytes) | Largest size ever reached by the queue (bytes) |
| Packets Marked ECN | Total number of packets marked by ECN capable network. |
| Total Packets Dropped Forcefully | Total packets dropped forcefully due to reasons other than the queue being full, such as the next hop not being reachable. |

Note: The average queue length is calculated as follows:

$$\text{average queue length} = \frac{\text{sum of weighted packet sizes for all packets}}{\text{(simulation time, in seconds)}},$$

where

$$\text{weighted packet size} = (\text{packet size, in bytes}) * (\text{time spent by the packet in the queue, in seconds})$$

6.9.5 Scenarios Included in QualNet

The QualNet distribution includes several sample scenarios for the WRED queue. All scenarios are located in the directory `QUALNET_HOME/scenarios/developer/queue/wred`. [Table 6-44](#) lists the sub-directory where each scenario is located.

TABLE 6-44. WRED Scenarios Included in QualNet

| Scenario | Description |
|-----------------|--|
| wred | Shows how packets are handled by WRED queue when bottleneck condition arrives. |

6.9.6 References

1. "Empirical Study Of Buffer Management Schemes For Diffserv Assured Forwarding PHB" (www.sce.carleton.ca/faculty/lambadaris/recent-papers/162.pdf)

2. <http://www.cisco.com/warp/public/732/Tech/red/>

6.10 Weighted Round Robin (WRR) Scheduler

6.10.1 Description

The WRR Scheduler is an alternative scheduling discipline in QualNet. Weighted Round Robin also attempts to service each priority queue based on a percentage allocation of the total outgoing bandwidth of the link. However, it does not calculate a “Finish Time” (the time at which a packet would have been serviced given a hypothetical fluid server) for each packet, instead, it establishes a round-robin order with multiple turns for highly weighted priority queues.

6.10.2 Features and Assumptions

This section describes the implemented features, omitted features, assumptions and limitations of the WRR model.

6.10.2.1 Implemented Features

- Improved performance by insertion of service prioritization.
- Ensures traffic gets a fair share of allocated bandwidth.

6.10.2.2 Omitted Features

None.

6.10.2.3 Assumptions and Limitations

- Users can specify a weight for each queue, but it should be >0 and <1. If there is no weight specified for queues, weights for each queue are assigned internally according to their priority.

6.10.3 Command Line Configuration

To select the WRR scheduler, include the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] IP-QUEUE-SCHEDULER           WEIGHTED-ROUND-ROBIN
```

The scope of this parameter declaration can be Global, Node, Subnet, or Interface. See [Section 1.2.1.1](#) for a description of `<Qualifier>` for each scope.

WRR Parameters

WRR parameters are described in [Table 6-45](#). See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 6-45. WRR Parameters

| Parameter | Value | Description |
|--|--|---|
| QUEUE-WEIGHT <i>Optional</i> Scope: All Instances: priority | Real <i>Range:</i> [0.0 , 1.0] <i>Default:</i> See note below table. | Weight of the queue. Note: The weights for all queues on an interface should add up to 1. |
| INPUT-SCHEDULER-STATISTICS <i>Optional</i> Scope: All | List • YES • NO <i>Default:</i> NO | Indicates whether statistics are collected for input scheduler. |
| SCHEDULER-STATISTICS <i>Optional</i> Scope: All | List • YES • NO <i>Default:</i> NO | Indicates whether statistics are collected for output scheduler. |

Note: The default weight of each queue is the ((priority value of the queue)+ 1) / (sum of all the priority values for all the queues + number of queues).

6.10.4 GUI Configuration

This section describes how to configure Weighted Round Robin (WRR) Scheduler model in the GUI.

Configuring WFQ Parameters

To configure the Weighted Round Robin Scheduler parameters, perform the following steps:

1. Go to one of the following locations:
 - To set wireless subnet properties, go to **Wireless Subnet Properties Editor > Network Layer > Schedulers and Queues**.
 - To set properties for a specific node, go to **Node Properties Editor > Node Configuration > Network Layer > Schedulers and Queues**.
 - To set properties for a specific interface of a node, go to one of the following locations:
 - **Interface Properties Editor > Interfaces > Interface # > Network Layer > Schedulers and Queues**.
 - **Default Device Properties Editor > Interfaces > Interface # > Network Layer > Schedulers and Queues**.

In this section, we show how to configure WRR parameters for a specific node using the Default Device Properties Editor. Parameters can be set in the other properties editors in a similar way.

2. Set IP Output Queue Scheduler to Weighted Round Robin.

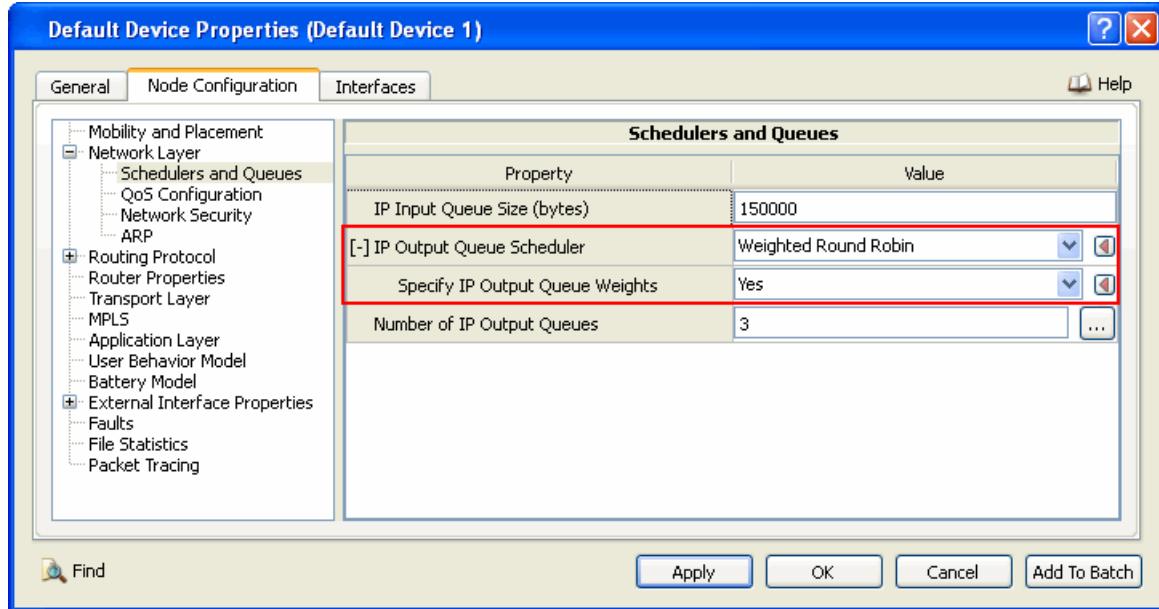


FIGURE 6-18. Setting WRR Parameters

Setting Parameters

- To specify IP Queue Weights, set **Specify IP Output Queue Weights** to Yes; otherwise, set **Specify IP Output Queue Weights** to No.
3. If **Specify IP Output Queue Weights** is set to Yes, then configure the queue weights as follows:
- Set **Number of IP Output Queues** to the desired value.
 - Click on the **Open Array Editor** button in the **Value** column. This opens the Array Editor.
 - In the left panel of the Array Editor, select the index of the queue to be configured. In the right panel, set the queue weight parameters listed in [Table 6-46](#) and the other parameters for the queue.

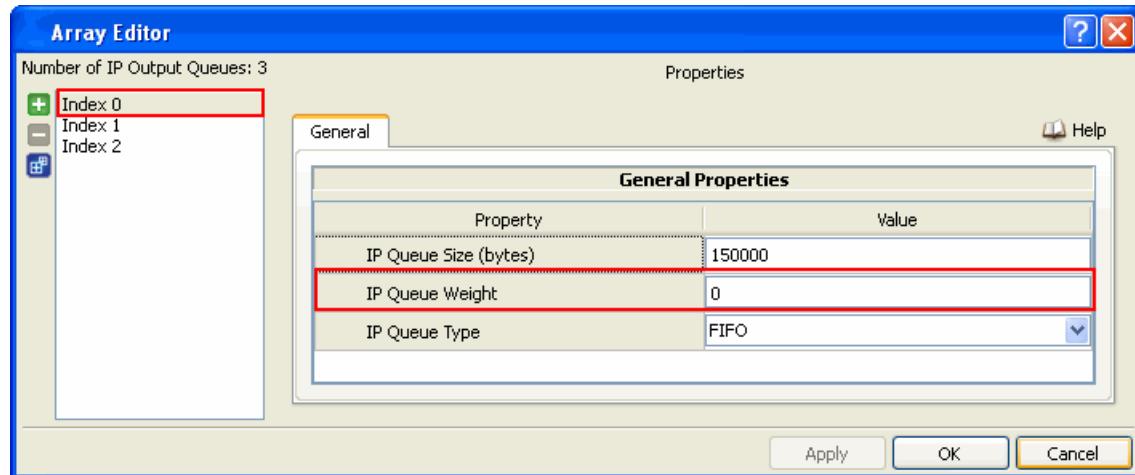


FIGURE 6-19. Setting Queue Weight Parameters

TABLE 6-46. Command Line Equivalent of Queue Weight Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-----------------|-------------------------|------------------------|
| IP Queue Weight | Node, Subnet, Interface | QUEUE-WEIGHT |

Configuring Statistics Parameters

Statistics for the Weighted Round Robin scheduler can be collected at the global, node, subnet, and interface levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for IP Input Queue Scheduler, check the box labeled **IP Input Scheduler** in the appropriate properties editor.

To enable statistics collection for IP Output Queue Scheduler, check the box labeled **IP Output Scheduler** in the appropriate properties editor.

TABLE 6-47. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------------|---------------------------------|----------------------------|
| IP Input Scheduler | Global, Node, Subnet, Interface | INPUT-SCHEDULER-STATISTICS |
| IP Output Scheduler | Global, Node, Subnet, Interface | SCHEDULER-STATISTICS |

6.10.5 Statistics

[Table 6-48](#) lists the WRR statistics that are output to the statistics (.stat) file at the end of simulation:

TABLE 6-48. WRR Statistics

| Statistics | Description |
|------------------|--|
| Packets Queued | Total number of packets enqueued in the queue |
| Packets Dequeued | Total number of packets dequeued from the queue |
| Packets Dropped | Total number of packets dropped from the queue |
| Service Ratio | Total number of packets dequeued to total number of dequeue requests |

6.10.6 Scenarios Included in QualNet

The QualNet distribution includes several sample scenarios for the WRR model. All scenarios are located in the directory `QUALNET_HOME/scenario/developer/wrr`. [Table 6-49](#) lists the sub-directory where each scenario is located.

TABLE 6-49. WRR Scenarios Included in QualNet

| Scenario | Description |
|----------|---|
| wrr | Shows the behavior of WRR scheduler at the intermediate node. |

6.10.7 References

The QualNet WRR model is based on the information available at the following URLs:

1. A Methodology for Study of Network Processing Architectures. Suryanarayanan, Deepak.(Under the direction of Dr.Gregory T Byrd.).
2. <http://www.lib.ncsu.edu/etd/public/etd-45401520610152001/etd.pdf>.

7

Transport Layer Models

This chapter describes features, configuration requirements and parameters, statistics, and scenarios for Transport Layer Models, and consists of the following sections:

- Abstract Transmission Control Protocol (Abstract TCP)
- Multicast Dissemination Protocol (MDP)
- Transmission Control Protocol (TCP)
- User Datagram Protocol (UDP)

7.1 Abstract Transmission Control Protocol (Abstract TCP)

7.1.1 Description

Abstract TCP is based on TCP Reno (see [Section 7.3](#)). However, it simplifies and omits some features in order to improve the runtime performance of the TCP model. Because of this, it has slightly lower fidelity as a tradeoff for simulation speed.

7.1.2 Features and Assumptions

This section describes the implemented features, omitted features, assumptions and limitations of the Abstract TCP model.

7.1.2.1 Implemented Features

The Abstract TCP model implements the TCP Reno features except the ones listed in [Section 7.1.2.2](#).

7.1.2.2 Omitted Features

The Abstract TCP model omits the following features:

- Three-way hand shaking (open/close)
- Keep alive
- Persistence time
- Variable packet size
- Actual data caching
- TCP options

7.1.2.3 Assumptions and Limitations

None.

7.1.3 Command Line Configuration

To enable Abstract TCP include the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] TCP ABSTRACT-TCP
```

The scope of this parameter declaration can be Global or Node. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

Abstract TCP Parameters

Table 7-1 shows the configuration parameters for Abstract TCP. See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 7-1. Abstract TCP Parameters

| Parameter | Value | Description |
|--|--|--|
| TCP-Delay-ACKS <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> YES | Specifies whether ACKs for data segment packets are delayed for a certain period of time to reduce the number of ACKs transmitted. If this parameter is set to NO, ACKs are sent immediately. |
| TCP-MSS <i>Optional</i> Scope: Global, Node | Integer <i>Range:</i> > 64 <i>Default:</i> 512 <i>Unit:</i> bytes | Specifies the maximum segment size. |
| TCP-RECEIVE-BUFFER <i>Optional</i> Scope: Global, Node | Integer <i>Range:</i> [1, 65535] <i>Default:</i> 16384 <i>Unit:</i> bytes | Specifies the receive buffer size. The receive buffer size provide an upper bound on the advertised window. The TCP sender cannot send more data than the available space of the receive buffer at the receiver. |
| TCP-SEND-BUFFER <i>Optional</i> Scope: Global, Node | Integer <i>Range:</i> > TCP-MSS <i>Default:</i> 16384 <i>Unit:</i> bytes | Specifies the send buffer size. The send buffer size provides an upper bound on the advertised window. The TCP sender cannot send more data than what is stored in the sender buffer. |
| TCP-STATISTICS <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | This parameter enables or disables TCP statistics collection. |
| TCP-TRACE <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• TCPDUMP-ASCII | This parameter enables TCP dump. Abstract TCP supports TCP dump in TCP Dump (ASCII) format only. The name of the dump file is tcptrace_abs.asc. |

TABLE 7-1. Abstract TCP Parameters (Continued)

| Parameter | Value | Description |
|---|--|--|
| TCP-TRACE-DIRECTION <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• OUTPUT• INPUT• BOTH <i>Default:</i> BOTH | Specifies the type(s) of packets that are included in the dump file. INPUT : Only packets received from the network are recorded in the dump file. OUTPUT: Only packets sent to the network are recorded in the dump file. BOTH : Both sent and received packets are recorded in the dump file. |
| TCP-VERIFICATION-DROP-COUNT <i>Optional</i> Scope: Global, Node | Integer <i>Range:</i> [0, 4] <i>Default:</i> 0 | Specifies the drop count. This option is for use with TCP verification scenarios that test variation in behavior with 0 to 4 drops. Note: This parameter is for testing purposes. It is recommended that the default value be used for this parameter. |

7.1.4 GUI Configuration

This section describes how to configure Abstract TCP in the GUI.

Configuring Abstract TCP Parameters

To configure the Abstract TCP parameters, perform the following steps:

1. Go to **Default Device Properties Editor > Node Configuration > Transport Layer**.
2. Set **Configure TCP** to Yes and set **TCP Variant** to *Abstract TCP*.

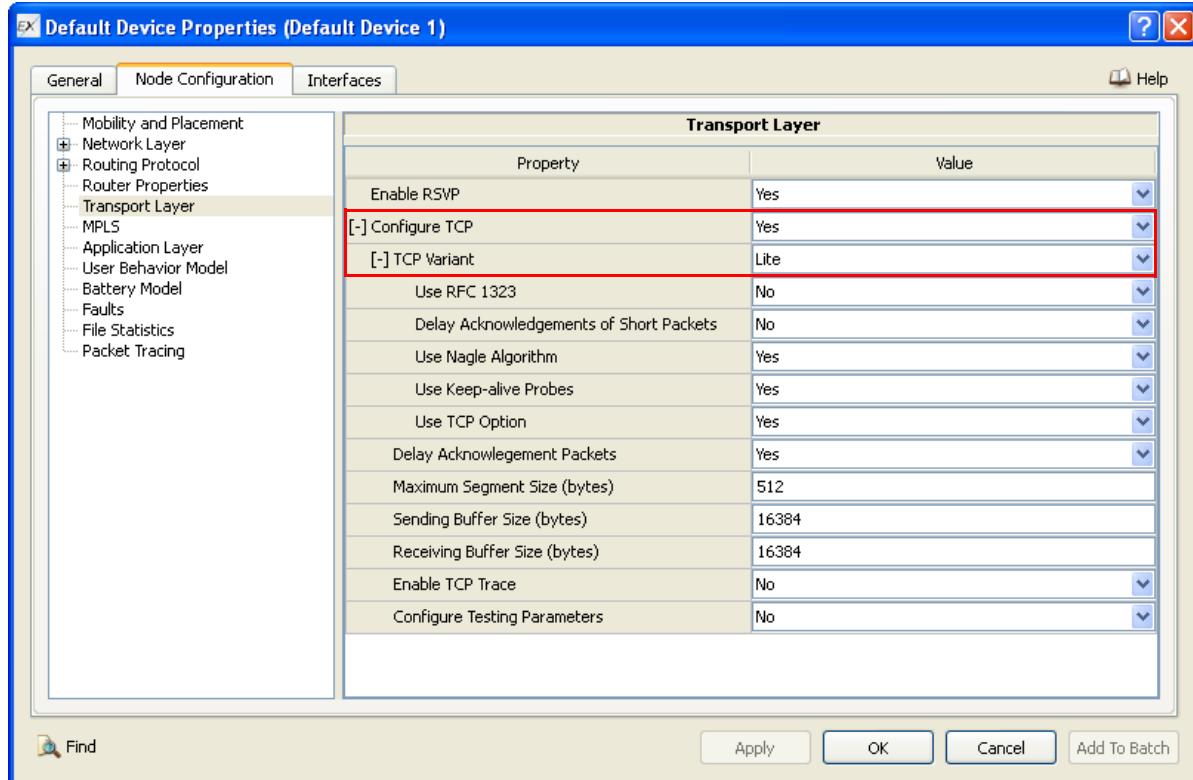


FIGURE 7-1. Configuring TCP Variant

TABLE 7-2. Command Line Equivalent of TCP Variant Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| TCP Variant | Node | TCP |

3. Set the dependent parameters for Abstract TCP listed in Table 7-3.

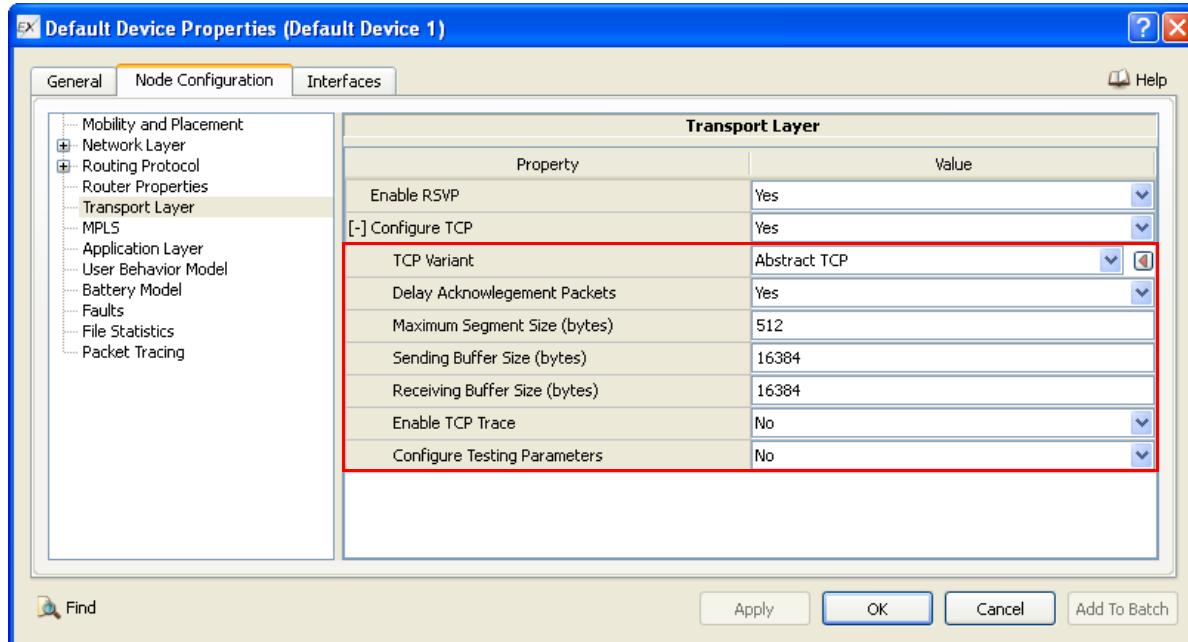


FIGURE 7-2. Setting Abstract TCP Parameters

TABLE 7-3. Command Line Equivalent of Abstract TCP Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-------------------------------|------------------------|------------------------|
| Delay Acknowledgement Packets | Node | TCP-DELAY-ACKS |
| Maximum Segment Size | Node | TCP-MSS |
| Sending Buffer Size | Node | TCP-SEND-BUFFER |
| Receiving Buffer Size | Node | TCP-RECEIVE-BUFFER |
| Enable TCP trace | Node | N/A |
| Configure Testing Parameters | Node | N/A |

4. To enable TCP trace, set **Enable TCP Trace** to Yes and set the dependent parameters listed in Table 7-4.

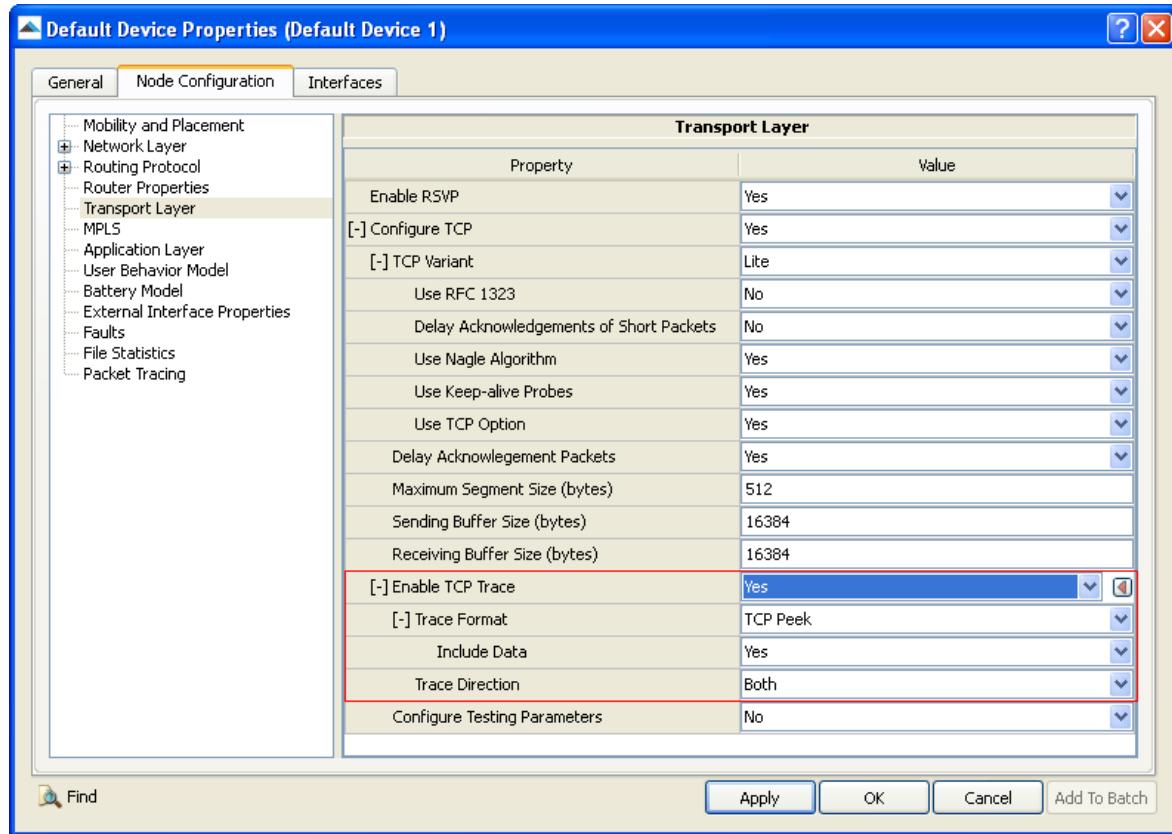


FIGURE 7-3. Enabling TCP Trace

TABLE 7-4. Command Line Equivalent TCP Trace Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-----------------|------------------------|------------------------|
| Trace Format | Node | TCP-TRACE |
| Include Data | Node | TCP-TRACE-WITH-DATA |
| Trace Direction | Node | TCP-TRACE-DIRECTION |

5. To set testing parameters, set the **Configure Testing Parameters** to Yes and set the dependent parameters shown in the Table 7-5.

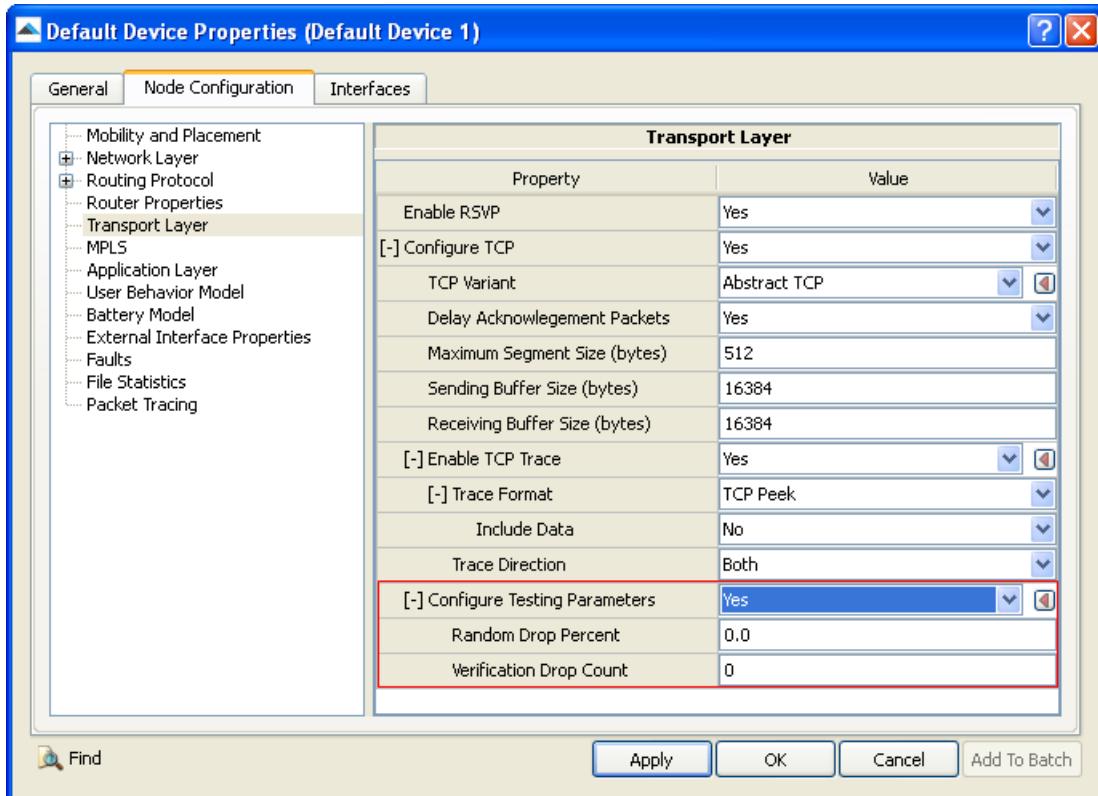


FIGURE 7-4. Configure Testing Parameters

TABLE 7-5. Command Line Equivalent TCP Configure Testing Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-------------------------|------------------------|-----------------------------|
| Random Drop Percent | Node | TCP-RANDOM-DROP-PERCENT |
| Verification Drop Count | Node | TCP-VERIFICATION-DROP-COUNT |

Configuring Statistics Parameters

Statistics for Abstract TCP can be collected at the global and node levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for Abstract TCP, check the box labeled **TCP** in the appropriate properties editor.

TABLE 7-6. Command Line Equivalent of TCP Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| TCP | Global, Node | TCP-STATISTICS |

7.1.5 Statistics

[Table 7-7](#) lists the Abstract TCP statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 7-7. Abstract TCP Statistics

| Statistic | Description |
|---|--|
| ACK-only Packets Sent | Total number of pure ACK packets sent to network layer from transport layer. |
| Data Packets Fast Retransmitted | Total number of packets which are fast retransmitted. |
| Data Packets in Sequence | Total number in sequence data packets sent. |
| Data Packets Received | Total number of pure data packets received at transport layer. |
| Data Packets Retransmitted | Total number of data packets retransmitted. |
| Data Packets Sent | Total number of pure data packets sent to network layer from transport layer. |
| Duplicate ACK Packets Received | Total number of in duplicated ACK packets received. |
| In Sequence ACK Packets Received | Total number of in sequence ACK packets received by the server. |
| Packets Received that are Too Short | Total number of short packets received with packet size smaller than TCP/IP header. |
| Packets Sent to Network Layer | Total number of packets sent to network layer from transport layer. It is the sum of data packets and control packets and pure ACK packets. But Abstract TCP has no control packets. |
| Total Packets Received From Network Layer | Total number of packets received from network layer to transport layer. It is the sum of data packets and control packets and pure ACK packets. But Abstract TCP has no control packets. |

7.1.6 Scenarios Included in QualNet

The QualNet distribution includes several sample scenarios for the Abstract TCP model. All scenarios are located in the directory `QUALNET_HOME/scenarios/developer/tcp`. [Table 7-8](#) lists the sub-directory where each scenario is located.

TABLE 7-8. Abstract TCP Scenarios Included in QualNet

| Scenario | Description |
|---|---|
| <code>abstract-tcp-0drop</code> | Shows the behaviour of Abstract TCP. |
| <code>abstract-tcp-1drop</code> | Shows the behaviour of Abstract TCP when one packet is dropped. |
| <code>abstract-tcp-2drop</code> | Shows the behaviour of Abstract TCP when 2 packets are dropped. |
| <code>abstract-tcp-3drop</code> | Shows the behaviour of Abstract TCP when 3 packets are dropped. |
| <code>abstract-tcp-4drop</code> | Shows the behaviour of Abstract TCP when 4 packets are dropped. |
| <code>abstract-tcp-delayed-ack-even-packet</code> | Shows the behaviour of Abstract TCP when delay ACK is ON with even number of application packets and time out has occurred. |
| <code>abstract-tcp-delayed-ack-odd-packet</code> | Shows the behaviour of Abstract TCP when delay ACK is ON with odd number of application packets. |

7.2 Multicast Dissemination Protocol

7.2.1 Description

The Multicast Dissemination Protocol (MDP) is a transport protocol which allows data to be reliably transmitted from a source (server) to a set of receivers on top of the generic UDP/IP multicast transport. MDP is well suited for reliable multicast bulk transfer of data across a heterogeneous inter-network. At its core, it is an efficient negative acknowledgement (NACK) based reliable multicast protocol and also includes an optional adaptive end-to-end rate-based congestion control mode.

The MDP framework consists of both receiver and source software modules. Multicast receivers wishing to subscribe to a multicast file dissemination service require the receiver module, while dissemination sources require the source module functionality.

MDP mainly uses selective negative acknowledgement (NACK) of missing data by receivers (clients) to enforce reliability. The NACK approach is used for maintaining multicast protocol efficiency and scalability. Redundant NACK transmissions are suppressed among a group of receivers using probabilistic techniques.

In conjunction with selective NACK, the MDP model uses a parity-based repairing mechanism based upon packet-level forward error correction coding concepts. The use of parity-based erasure repairing for multicast selective retransmission offers significant performance advantages (especially in error-prone wireless environments or across scaled WAN sessions). In MDP, encoded parity repair packets are normally sent only in response to repair requests by receivers. Therefore, the algorithm adds no additional protocol overhead above pure selective retransmission methods. However, the protocol may be optionally configured to transmit "proactive" repair packets as part of the original data transmission block.

In addition to its pure NACK-based mode of operation, MDP provides an optional mode for a source to request positive acknowledgment (ACK) or receipt of individual transport objects from a specific set of receivers in the group. The list of receivers providing receipt acknowledgement is determined by the server application with a priori knowledge of participating nodes and/or by receivers which indicate an ACKing status with a flag in their MDP_REPORT messages. Positive acknowledgment can be requested for all transport objects sent by the server or may be applied at certain "watermark" progress points in the course of transmission of a series (stream) of transport objects.

The MDP allows senders and receivers to dynamically join and leave multicast sessions with a minimal amount of overhead for control information and timing synchronization among participants. As a result of this requirement, MDP protocol message headers contain some common information allowing receivers to easily synchronize with sources on a dynamic, ad hoc basis. In its common mode of operation, the MDP protocol uses multicast delivery mechanisms for both source and receiver transmissions, but the protocol permits optional unicast-based client feedback to MDP data sources. Optional unicast feedback may be suitable for use in asymmetric networks or in networks where only unidirectional multicast routing/delivery service exists.

MDP also provides some useful protocol features, such as Emission Controlled (EMCON) or "silent" mode of client operation. In EMCON mode, receiving nodes are able to receive messages but they are not able to acknowledge the reception of these messages. When it is not possible or desirable for the client nodes to transmit messages back to the server node, or if delivery delay is of paramount concern, the protocol can provide proactive parity data for more effective unidirectional repairing. This is an optional feature which can be configured in the model. The MDP model provides support for basic EMCON modes of operation.

MDP is used under a variety of heterogeneous network architectures. Targeted operational factors that are key considerations for MDP are:

- Use in heterogeneous, WAN infrastructures
- Use in mobile and wireless network conditions
- Operation in asymmetric delivery scenarios
- Support for small to large group sizes
- Support for group dynamics

7.2.2 Features and Assumptions

This section describes the implemented features, omitted features, assumptions and limitations of the MDP model.

7.2.2.1 Implemented Features

- One-to-many bulk data multicast dissemination
- Selective negative acknowledgement (NACK)
- NACK backoff to avoid receiver message implosion
- Aggregation of control messaging for bandwidth efficiency
- Automated group round-trip timing
- Support for asymmetric operation
- Parity-based repair mechanism
- Adaptive end-to-end rate-based congestion control
- Optional positive receipts from selected receivers
- Support for EMCON (silent clients) mode
- Support for Data Object transmission in simulation as well as emulation mode
- Support for File Object transmission in simulation as well as emulation mode

7.2.2.2 Omitted Features

None.

7.2.2.3 Assumptions and Limitations

- The QualNet MDP model conforms to the Navy Research Laboratory (NRL) implementation of MDP (release *src-mdp-2.1b6*). In particular, "Silent" receiver or EMission CONtrolled (EMCON) receiver operation and stream integrity operation are subject to the limitations of the NRL implementation.
- MDP Data Object transmission in simulation mode is supported only for the following applications:
 - Constant Bit Rate (CBR)
 - Forward Application Traffic Generator (Forward-App), which is a special application for use only with external interfaces
 - Multicast Constant Bit Rate Traffic Generator (MCBR)
 - Super Application Traffic Generator (Super-App) in single-host and client-server modes
 - Traffic Generator (Traffic-Gen) in single-host and client-server modes
 - Trace File-based Traffic Generator (Traffic-Trace) in single-host and client-server modes
 - Variable Bit Rate (VBR)

7.2.3 Supplemental Information

In MDP, each packet is treated as a single object with a unique object ID. For better performance, multicast applications should be configured with a large packet size and a long enough inter-arrival time so that it can simulate the behavior of bulky data.

7.2.4 Command Line Configuration

Running an application with MDP requires two steps:

- **Defining MDP Profiles:** An MDP profile specifies the values of a set of MDP session control parameters. MDP profiles are defined in the MDP profile file. If no MDP profiles are defined, then default values for MDP session control parameters are used for an MDP-enabled application.
- **Enabling MDP and Specifying MDP Profile:** MDP can be enabled for a specific application in the application configuration file. To run Forward-App with MDP, MDP must be also enabled at the node. Similarly, the MDP profile to be used with an application session can be specified in the application configuration file or a default MDP profile can be associated with the node.

Parameters for configuring MDP are specified in the following files:

- **Scenario Configuration File:** Parameters for enabling MDP and for associating default MDP profiles with nodes are specified in the scenario configuration (.config) file.
- **Application Configuration File:** To use an application with MDP, MDP-specific parameters are added in the application configuration (.app) file. See the description of the applications that support MDP for details.
- **MDP Profile File:** Different MDP profiles are defined using the MDP profile file.

7.2.4.1 MDP Parameters for the Scenario Configuration File

To enable MDP, include the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] MDP-ENABLED      YES
```

The scope of this parameter declaration can be Global or Node. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

- Notes:**
1. The default value of the parameter MDP-ENABLED is NO.
 2. For running MDP with CBR, MCBR, Super-App, Traffic-Gen, Traffic-Trace, or VBR, MDP does not have to be explicitly enabled at the source node (instead, parameter MDP-ENABLED can be specified with the other application parameters in the application configuration file). However, to run MDP with Forward-App, MDP must be explicitly enabled at the node using the parameter MDP-ENABLED in the scenario configuration (.config) file.

[Table 7-9](#) lists the MDP parameters that can be specified in the scenario configuration file. See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 7-9. MDP Parameters

| Parameter | Value | Description |
|--|--|---|
| MDP-PROFILE-FILE <i>Optional</i> Scope: Global, Node | Filename | <p>Name of the MDP profile file.</p> <p>The MDP profile file defines MDP profiles to use with applications. If this file is not defined, then default values for MDP profile parameters are used if MDP is enabled.</p> <p>The format of the MDP profile file is described in Section 7.2.4.3. This file usually has the extension “mdp”.</p> |
| MDP-PROFILE <i>Optional</i> Scope: Global, Node | String | <p>Name of the default MDP profile to be used with MDP-enabled applications running at the node.</p> <p>This should be the name of a MDP profile defined in the MDP profile file (see Section 7.2.4.3).</p> <p>Note: The default MDP profile can be overwritten by specifying a different MDP profile with an application in the application configuration (.app) file (see Section 7.2.4.2).</p> |
| MDP-TX-CACHE <i>Optional</i> Scope: Global, Node | <p>Triplet of integer values (see description)</p> <p><i>Range:</i> ≥ 0</p> <p><i>Default:</i> See description</p> | <p>Parameters for transmit cache depth.</p> <p>These are the default values for the transmit cache depth parameters for all profiles. If transmit cache depth parameters are not specified for a profile in the MDP Profile file by means of the parameter TX-CACHE (see Section 7.2.4.3), then these values are used for that profile.</p> <p>The transmit cache depth affects how long the application retains state on previously transmitted data objects for possible repair transmission.</p> <p>This parameter is specified in the following format:</p> <p>MDP-TX-CACHE <min-cnt> <max-cnt> <max-sz></p> <p>where</p> <ul style="list-style-type: none"> <min-cnt> : Smallest number of data objects kept (regardless of size). The default value is 8. <max-cnt> : Largest number of data objects kept (regardless of size). The default value is 5000. <max-sz> : Maximum size, in bytes. The default value is 8388608 (= $8 \times 1024 \times 1024$). <p>The amount of history stored is limited by <max-cnt> and <max-sz>, whichever limit is reached first.</p> |
| MDP-TX-BUFFER-SIZE <i>Optional</i> | <p>Integer</p> <p><i>Range:</i> ≥ 0</p> <p><i>Default:</i> 1048576</p> <p><i>Unit:</i> bytes</p> | <p>Size of the server's transmitting buffer.</p> <p>This is the default value for the server's transmitting buffer size for all profiles. If the server's transmitting buffer size is not specified for a profile in the MDP Profile file by means of the parameter TX-BUFFER-SIZE (see Section 7.2.4.3), then this value is used for that profile.</p> |

TABLE 7-9. MDP Parameters (Continued)

| Parameter | Value | Description |
|--|---|--|
| MDP-RX-BUFFER-SIZE <i>Optional</i> | Integer <i>Range:</i> ≥ 0 <i>Default:</i> 1048576 <i>Unit:</i> bytes | Amount of memory the receiver allocates for buffering each server heard. This is the default value for the amount of allocated memory for all profiles. If the amount of allocated memory is not specified for a profile in the MDP Profile file by means of the parameter RX-BUFFER-SIZE (see Section 7.2.4.3), then this value is used for that profile. |
| MDP-SEGMENT-POOL-SIZE <i>Optional</i> | Integer <i>Range:</i> ≥ 0 <i>Default:</i> 10 | Parameter that determines the amount of pre-allocated memory for MDP data and parity segments. The amount of pre-allocated memory increases with the value of this parameter. This is the default value for the memory pre-allocation parameter for all profiles. If the memory pre-allocation parameter is not specified for a profile in the MDP Profile file by means of the parameter SEGMENT-POOL-SIZE (see Section 7.2.4.3), then this value is used for that profile. Note: A smaller value of this parameter will result in memory saving because less memory is pre-allocated for MDP data and parity segments, but it may affect the runtime performance. |
| MDP-STATISTICS <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Specifies whether statistics are collected for MDP. |

7.2.4.2 Running Forward-App with MDP

Forward Application Traffic Generator (Forward-App) is a special application for use only with external interfaces. In order to run Forward-App with MDP at a node, MDP must explicitly be enabled at the node by using the parameter MDP-ENABLED in the scenario configuration file.

If an MDP profile file is not specified in the scenario configuration file or a default MDP profile is not associated with the node, then default values for MDP session control parameters are used for the Forward-App session.

7.2.4.3 Format of the MDP Profile File

The MDP profile file defines one or more MDP profiles to use with applications. A MDP profile is a set of MDP session control parameters which specify the model behavior. If this parameter is not defined then default values of MDP related parameters are used. The normal extension of the MDP profile file is “mdp”.

The first line of a MDP profile assigns a name to the profile and has the following format:

```
MDP-PROFILE <profile-name>
```

where

| | |
|----------------|--------------------------|
| <profile-name> | Name of the MDP profile. |
|----------------|--------------------------|

The MDP profile name is followed by the parameters that define the session control characteristics associated with the profile. These parameters are described in [Table 7-10](#).

- Notes:**
1. All parameters in [Table 7-10](#) are optional. If a parameter is not specified, its default value is used in the MDP profile.
 2. Each of the parameters in [Table 7-10](#) should occur at most once in a MDP profile. In case of multiple occurrences, the last configured value applies to that profile.
 3. Each parameter must be entered on a single line by itself without any prefix.
 4. The profile parameters can be entered in any order after the profile name.

TABLE 7-10. MDP Profile Parameters

| Parameter | Value | Description |
|---------------------------------|--|--|
| TOS <i>Optional</i> | Integer <i>Range:</i> [0, 255] <i>Default:</i> 0 | Value of the IP type-of-service (TOS) header byte. |
| TX-RATE <i>Optional</i> | Integer <i>Range:</i> ≥ 0 <i>Default:</i> 64000 <i>Unit:</i> bps | Maximum transmission rate in bits per second. |
| TTL-COUNT <i>Optional</i> | Integer <i>Range:</i> [0, 255] <i>Default:</i> 8 | Multicast time-to-live (hop count) for scoping. |
| INITIAL-GRTT <i>Optional</i> | Real <i>Range:</i> [0.001, 15.0] <i>Default:</i> 0.5 <i>Unit:</i> seconds | Initial estimate of Greatest Round Trip Time (GRTT). This option controls the server's initial assumption of GRTT before hearing from any of its clients. |
| SEGMENT-SIZE <i>Optional</i> | Integer <i>Range:</i> [64, 8128] <i>Default:</i> 1024 <i>Unit:</i> bytes | Sender's maximum/normal segment (packet payload) size. |
| BLOCK-SIZE <i>Optional</i> | Integer <i>Range:</i> [1, 255] <i>Default:</i> 64 | Number of data segments (packets) per MDPv2 coding block. |

TABLE 7-10. MDP Profile Parameters (Continued)

| Parameter | Value | Description |
|---|---|--|
| NUM-PARITY <i>Optional</i> | Integer <i>Range:</i> [0, 128] <i>Default:</i> 32 | Number of available parity segments per MDPv2 coding block. Note: This parameter should be set such that $\text{BLOCK-SIZE} + \text{NUM-PARITY} \leq 255$. |
| NUM-AUTO-PARITY <i>Optional</i> | Integer <i>Range:</i> [0, NUM-PARITY] <i>Default:</i> 0 | Number of parity packets automatically sent by the server per MDP block. |
| EMCON-ENABLED <i>Optional</i> | List: • YES • NO <i>Default:</i> NO | Indicates whether emission control (EMCON) mode is enabled. Note: It is recommended that the flow/congestion control algorithm be disabled if the EMCON mode is enabled. |
| LOOPBACK-ENABLED <i>Optional</i> | List: • YES • NO <i>Default:</i> NO | Indicates whether multicast loopback is enabled. This allows a client to receive its own transmissions. |
| REPORT-MESSAGES-ENABLED <i>Optional</i> | List: • YES • NO <i>Default:</i> NO | Indicates whether transmission of status reports (MDP_REPORT messages) is enabled. |
| POSITIVE-ACK-WITH-REPORT <i>Optional</i> | List: • YES • NO <i>Default:</i> NO | Indicates whether positive acknowledgements are sent when REPORT-MESSAGES-ENABLED is set to YES. |
| POSITIVE-ACK-NODES <i>Optional</i> | List of IPv4 Addresses (see description) | IPv4 addresses of receiver nodes for which the sender requests positive acknowledgments. The list of IPv4 addresses can be separated by commas, spaces, or tabs. |
| UNICAST-NACK-ENABLED <i>Optional</i> | List: • YES • NO <i>Default:</i> NO | Indicates whether the receiver sends unicast protocol messages back to the sender. If this parameter is set to NO, then multicast NACKs are enabled. |
| MULTICAST-ACK-ENABLED <i>Optional</i> | List: • YES • NO <i>Default:</i> NO | Indicates whether the client multicasts ACK messages. By default, ACKs are sent as unicast messages to the sender. |

TABLE 7-10. MDP Profile Parameters (Continued)

| Parameter | Value | Description |
|---|---|--|
| BASE-OBJECT-ID <i>Optional</i> | Integer <i>Range:</i> ≥ 0 <i>Default:</i> 1 | Indicates the server's initial transmit object ID. |
| ARCHIVE-MODE-ENABLED <i>Optional</i> | List: • YES • NO <i>Default:</i> NO | Indicates whether the client should permanently store files instead of using the archive directory just as a temporary cache for received files. |
| ARCHIVE-PATH <i>Optional</i> | String <i>Default:</i> NULL | Path to the directory where the client stores (temporarily or permanently) received files. NULL refers to the current directory. |
| STREAM-INTEGRITY-ENABLED <i>Optional</i> | List: • YES • NO <i>Default:</i> YES | Indicates whether the MDP client requests repair of objects that are missed in their entirety or only of partially received objects. If this parameter is set to NO, the client requests repair of only partially received objects. If small files or messages are being transmitted, setting this parameter to NO may result in data being missed at the receivers. |
| FLOW-CONTROL-ENABLED <i>Optional</i> | List: • YES • NO <i>Default:</i> YES | Indicates whether experimental flow/congestion control algorithm is enabled. |
| FLOW-CONTROL-TX-RATE <i>Optional</i> | Pair of integer values (see description) <i>Range:</i> ≥ 0 <i>Default:</i> See description <i>Unit:</i> bps | Minimum and maximum transmission rates used by MDP's congestion control algorithm. This parameter is specified in the following format: FLOW-CONTROL-TX-RATE <min> <max> where <min> : Minimum transmission rate. <max> : Maximum transmission rate. If both are set to 0, the congestion control algorithm adjusts the rate to any value it determines. |

TABLE 7-10. MDP Profile Parameters (Continued)

| Parameter | Value | Description |
|--------------------------------------|---|---|
| TX-CACHE <i>Optional</i> | Triplet of integer values (see description) <i>Range:</i> ≥ 0 <i>Default:</i> MDP-TX-CACHE <i>Unit:</i> bytes | Parameters for transmit cache depth. This affects how long the application retains state on previously transmitted data objects for possible repair transmission. This parameter is specified in the following format: TX-CACHE <min-cnt> <max-cnt> <max-sz> where <min-cnt> : Smallest number of data objects kept (regardless of size). <max-cnt> : Largest number of data objects kept (regardless of size). <max-sz> : Maximum size, in bytes. The amount of history stored is limited by <max-cnt> and <max-sz>, whichever limit is reached first. |
| TX-BUFFER-SIZE <i>Optional</i> | Integer <i>Range:</i> ≥ 0 <i>Default:</i> MDP-TX-BUFFER-SIZE <i>Unit:</i> bytes | Size of the server's transmitting buffer. |
| RX-BUFFER-SIZE <i>Optional</i> | Integer <i>Range:</i> ≥ 0 <i>Default:</i> MDP-RX-BUFFER-SIZE <i>Unit:</i> bytes | Amount of memory the receiver allocates for buffering each server heard. |
| SEGMENT-POOL-SIZE <i>Optional</i> | Integer <i>Range:</i> ≥ 0 <i>Default:</i> MDP-SEGMENT-POOL-SIZE | Parameter that determines the amount of pre-allocated memory for MDP data and parity segments. The amount of pre-allocated memory increases with the value of this parameter. Note: A smaller value of this parameter will result in memory saving because less memory is pre-allocated for MDP data and parity segments, but it may affect the runtime performance. |
| ROBUSTNESS-COUNT <i>Optional</i> | Integer <i>Range:</i> ≥ 0 <i>Default:</i> 20 | Maximum probe count that should not be exceeded while resending MDP messages (such as MDP_CMD_ACK_REQ, MDP_CMD_FLUSH, etc.). |

7.2.5 GUI Configuration

To configure MDP in the GUI, properties have to be configured at the node level. Statistics parameters can be configured at both node and scenario levels. In addition, MDP has to be enabled for the applications.

Configuring MDP at a Node

To configure MDP at a node, do the following:

1. Go to **Default Device Properties Editor > Node Configuration > Application Layer**.
2. To enable MDP, set **Enable MDP** to Yes.

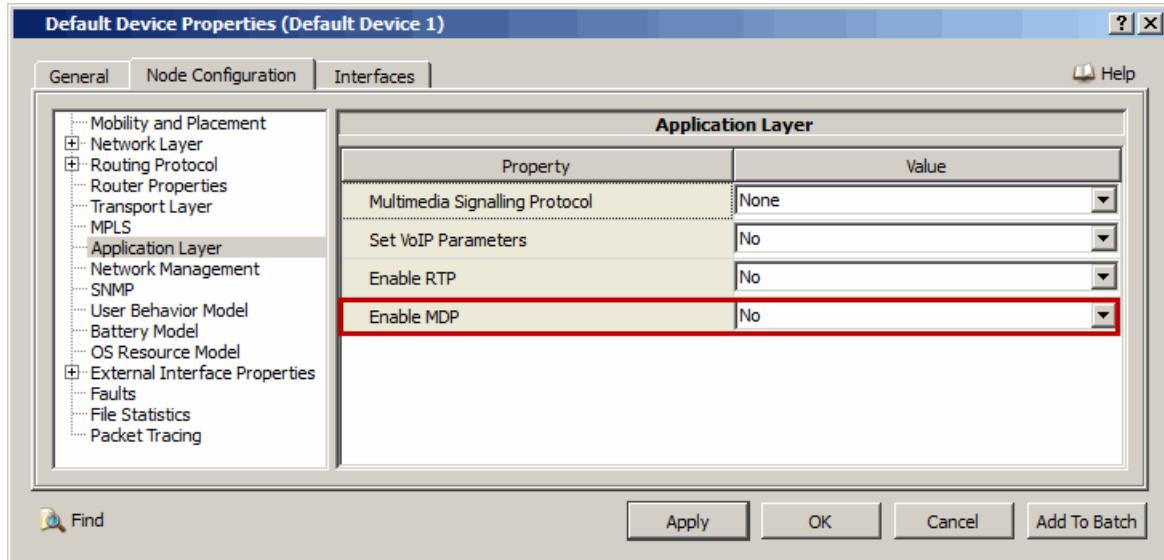


FIGURE 7-5. Enabling MDP

TABLE 7-11. Command Line Equivalent of Enabling MDP Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| Enable MDP | Node | MDP-ENABLED |

3. Set the MDP parameters listed in [Table 7-12](#).

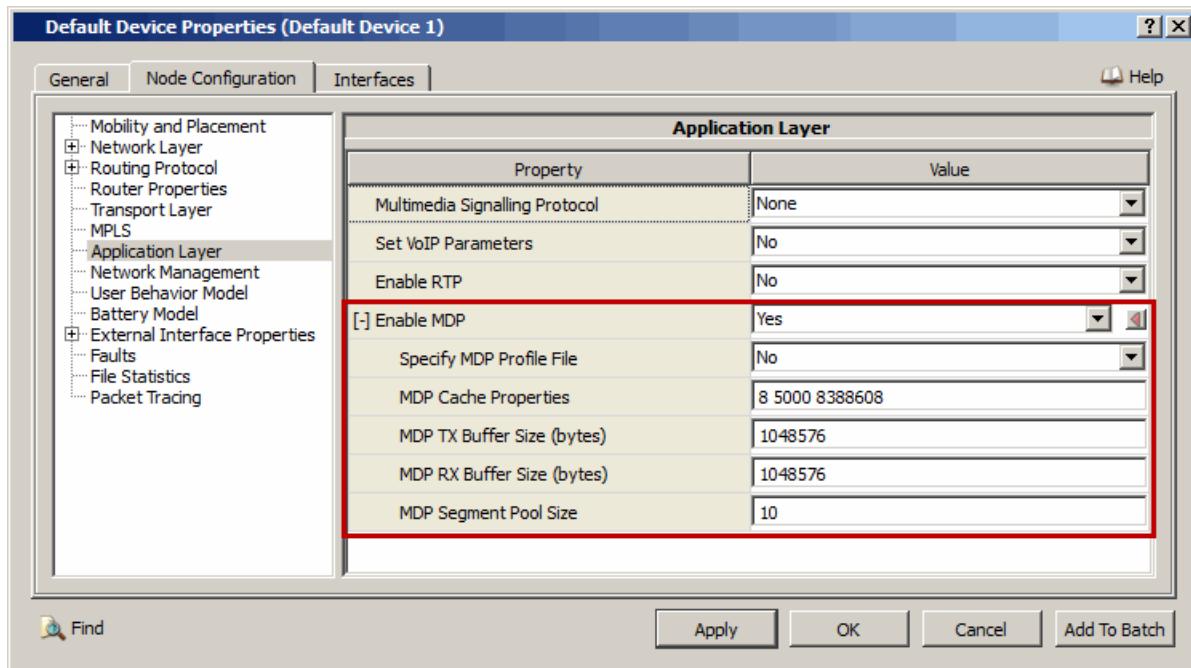


FIGURE 7-6. Setting MDP Parameters

TABLE 7-12. Command Line Equivalent of MDP Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|--------------------------|------------------------|------------------------|
| Specify MDP Profile File | Node | N/A |
| MDP TX Cache Properties | Node | MDP-TX-CACHE |
| MDP TX Buffer Size | Node | MDP-TX-BUFFER-SIZE |
| MDP RX Buffer Size | Node | MDP-RX-BUFFER-SIZE |
| MDP Segment Pool Size | Node | MDP-SEGMENT-POOL-SIZE |

Setting Parameters

- To specify a MDP Profile file, set **Specify MDP Profile File** to Yes; otherwise, set **Specify MDP Profile File** to No.

4. If **Specify MDP Profile File** is set to Yes, then set the dependent parameters listed in Table 7-13.

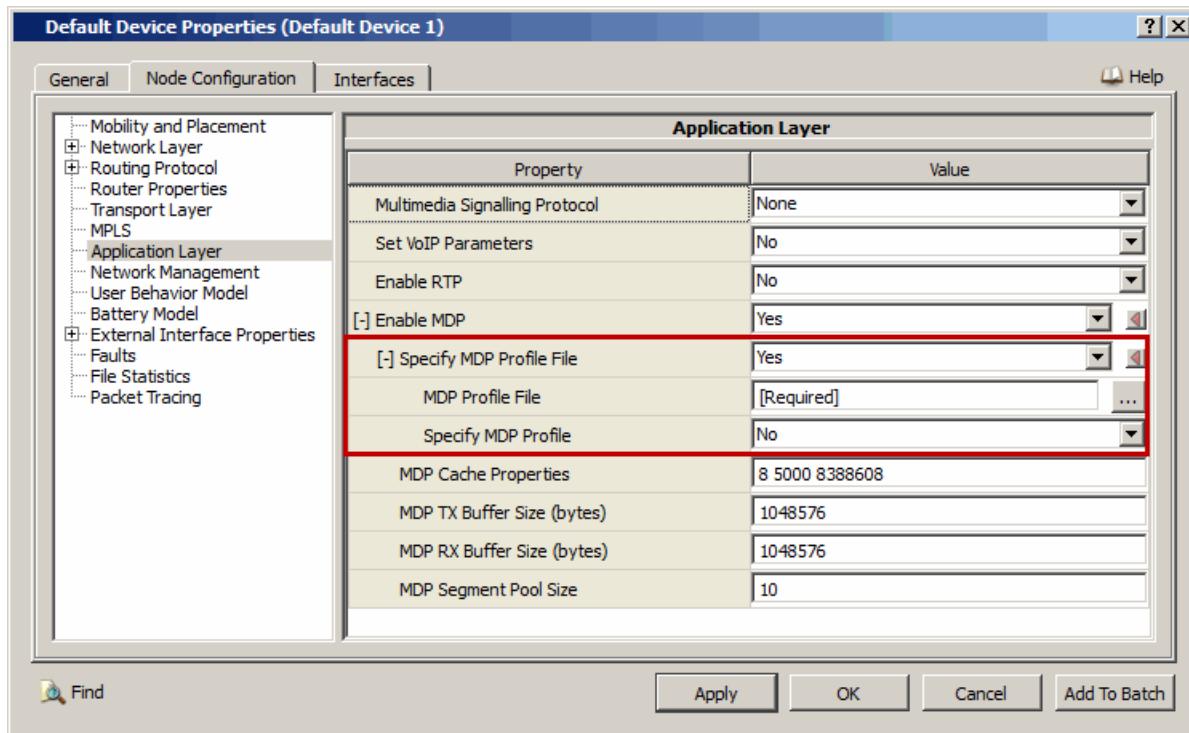


FIGURE 7-7. Specifying MDP Profile File

TABLE 7-13. Command Line Equivalent of MDP Profile File Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------------|------------------------|------------------------|
| MDP Profile File | Node | MDP - PROFILE - FILE |
| Specify MDP Profile | Node | N/A |

Setting Parameters

- To specify a default MDP profile for the node, set **Specify MDP Profile** to Yes; otherwise, set **Specify MDP Profile** to No.

5. If **Specify MDP Profile** is set to Yes, then set the dependent parameters listed in [Table 7-14](#).

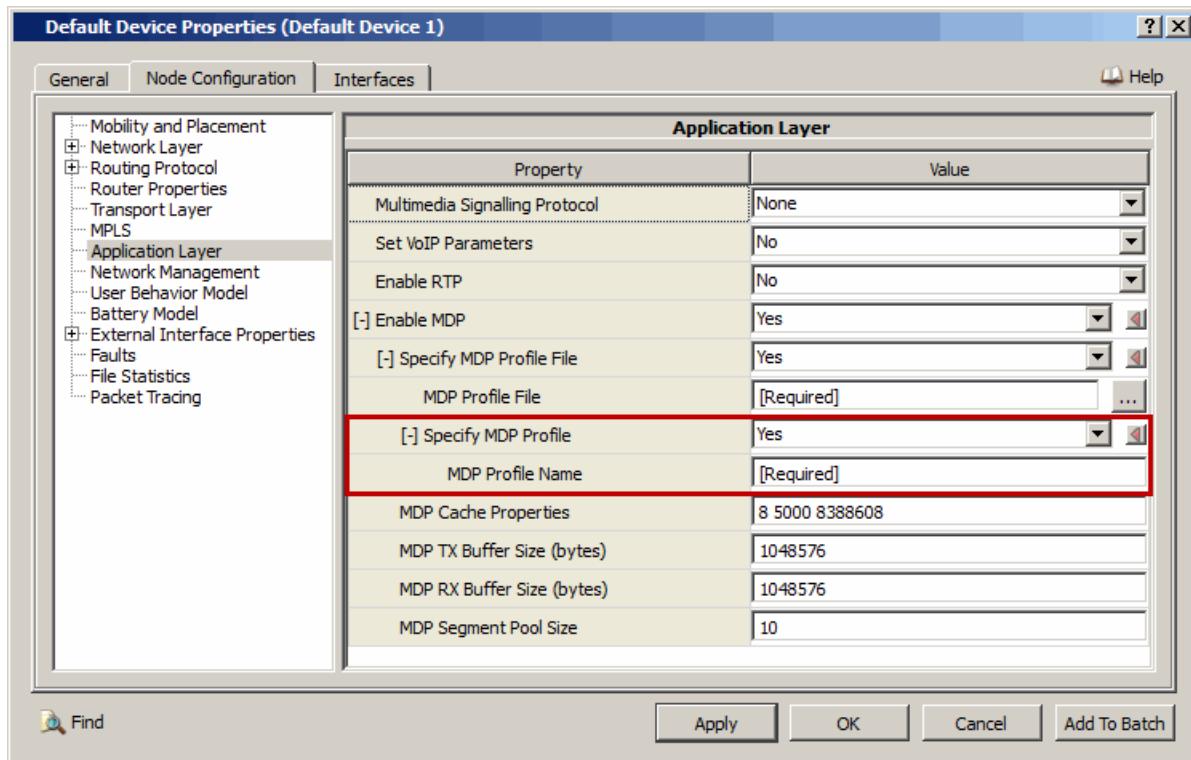


FIGURE 7-8. Specifying MDP Profile Name

TABLE 7-14. Command Line Equivalent of MDP Profile Name Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|------------------|------------------------|------------------------|
| MDP Profile Name | Node | MDP - PROFILE |

Configuring Statistics Parameters

Statistics for the MDP model can be collected at the global and node levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for MDP, check the box labeled **MDP** in the appropriate properties editor.

TABLE 7-15. Command Line Equivalent of MDP Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| MDP | Global, Node | MDP - STATISTICS |

7.2.6 Statistics

[Table 7-16](#) lists the statistics collected for the MDP model that are printed to the statistics file at the end of simulation.

Note: The statistics in [Table 7-16](#) are cumulative statistics for all MDP sessions at a node.

TABLE 7-16. MDP Statistics

| Statistic Collected | Description |
|--|---|
| Total Number of MDP Sessions | Total number of MDP sessions defined on node. |
| Total Number of Data Objects Fail to Queue | Total number of application data objects that failed to queue in MDP sessions. |
| Total Number of File Objects Fail to Queue | Total number of application file objects that failed to queue in MDP sessions. |
| Total MDP_CMD Messages Sent | Total number of command messages sent by the senders. |
| Total MDP_CMD Messages Received | Total number of command messages received by the receivers. |
| Total MDP_CMD_FLUSH Messages Sent | Total number of Flush command messages sent by the senders. |
| Total MDP_CMD_FLUSH Messages Received | Total number of Flush command messages received by the receivers. |
| Total MDP_CMD_SQUELCH Messages Sent | Total number of Squelch command messages sent by the senders. |
| Total MDP_CMD_SQUELCH Messages Received | Total number of Squelch command messages received by the receivers. |
| Total MDP_CMD_ACK_REQ Messages Sent | Total number of Positive Acknowledge command messages sent by the senders. |
| Total MDP_CMD_ACK_REQ Messages Received | Total number of Positive Acknowledge command messages received by the receivers. |
| Total MDP_ACK Messages Sent | Total number of response messages sent by the receivers for Positive ACK command. |
| Total MDP_ACK Messages Received | Total number of response messages received by the senders for Positive ACK command. |
| Total MDP_CMD_GRTT_REQ Messages Sent | Total number of GRTT request command messages sent by the senders. |
| Total MDP_CMD_GRTT_REQ Messages Received | Total number of GRTT request command messages received by the receivers. |
| Total Response Sent for MDP_CMD_GRTT_REQ | Total number of response messages sent by the receivers for GRTT command. |
| Total Response Received for MDP_CMD_GRTT_REQ | Total number of response messages received by the senders for GRTT command. |
| Total MDP_CMD_NACK_ADV Messages Sent | Total number of NACK Adv command messages sent by the senders. |
| Total MDP_CMD_NACK_ADV Messages Received | Total number of NACK Adv command messages received by the receivers. |
| Total MDP_NACK Messages Sent | Total number of NACK messages sent by the receivers. |
| Total MDP_NACK Messages Received | Total number of NACK messages received by the senders. |
| Total MDP_NACK Messages Suppressed | Total number of NACK messages suppressed by the receivers. |
| Total MDP_REPORT Messages Sent | Total number of Report messages sent by the receivers. |
| Total MDP_REPORT Messages Received | Total number of Report messages received by the senders. |

TABLE 7-16. MDP Statistics (Continued)

| Statistic Collected | Description |
|---|---|
| Total MDP_INFO Messages Sent | Total number of Info messages sent by the senders. |
| Total MDP_INFO Messages Received | Total number of Info messages received by the receivers. |
| Total MDP_DATA Messages Sent | Total number of Data messages sent by the senders. |
| Total MDP_DATA Messages Received | Total number of Data messages received by the receivers. |
| Total MDP_PARITY Messages Sent | Total number of Parity messages sent by the senders. |
| Total MDP_PARITY Messages Received | Total number of Parity messages received by the receivers. |
| Total Received MDP Objects Completed | Total number of received objects completed at the receivers. |
| Total Received MDP Objects Pending | Total number of received objects pending at the receivers. |
| Total Received MDP Objects Failed | Total number of received objects failed at the receivers. |
| Total Number of MDP DATA & PARITY Bytes Sent | Total number of data and parity bytes sent by the senders. |
| Total Number of MDP DATA & PARITY Bytes Received | Total number of data and parity bytes received by the receivers. |
| Total Number of MDP Blocks Received | Total number of blocks received by the receivers. |
| Total Number of Resynchronize with Sender Performed | Total number of times the resynchronization was done with sender by the receivers. |
| Total Number of MDP Block & Vector Pool Overruns | Total number of receiver's buffer usage overruns. |
| Maximum Usage of MDP Block & Vector Pool | Maximum usages of combine count for block and vector pool. (Here vector size is equal to segment size.) |

7.2.7 References

1. Joseph P. Macker, R. Brian Adamson, "The Multicast Dissemination Protocol (MDP)," draft-macker-rmt-mdp-00. October 1999.
2. Joseph P. Macker, R. Brian Adamson, "The Multicast Dissemination Protocol (MDP) ToolKit," IEEE Milcom. 1999.
3. Joseph P. Macker, R. Brian Adamson, "Reliable Multicast Congestion Control (RMCC)", IEEE Milcom. 2000.
4. Naval Research Laboratory implementation of MDP. <http://cs.itd.nrl.navy.mil/work/mdp>.

7.3 Transmission Control Protocol (TCP)

The QualNet TCP model is based on RFC 793, RFC 2481, and RFC 1323.

7.3.1 Description

The Transmission Control Protocol (TCP) is a highly reliable host-to-host protocol between hosts in packet-switched computer communication networks, and in interconnected systems of such networks. The TCP fits into a layered protocol architecture just above IP which provides a way for the TCP to send and receive variable-length datagrams.

QualNet's TCP model uses code converted from the FreeBSD 2.2.2 source code implementation for TCP. The differences between the QualNet code and the FreeBSD code are improvements for simulation performance, configurability through QualNet configuration files and GUI, interface code to the simulator, and modifications to offer additional variants of TCP. FreeBSD's TCP variant is known as "TCP Lite".

TCP Versions

There are several versions of TCP that you can specify via the QualNet configuration file: TCP Lite, TCP NewReno, TCP Reno, TCP SACK, and TCP Tahoe. These are modified in real systems by using compiler directives to enable sections of code, and then recompiling the kernel. Some operating systems also provide ways to dynamically modify some of the TCP parameters at runtime. In QualNet, change them by editing the configuration file. You can either modify these parameters individually, or you can select one of the well known TCP versions that will automatically activate the appropriate parameters. TCP Lite is the default variant of TCP for QualNet.

7.3.2 Features and Assumptions

This section describes the implemented features, omitted features, assumptions and limitations of the TCP model.

7.3.2.1 Implemented Features

- TCP Lite includes the following features:
 - Slow Start
 - Congestion Avoidance
 - Fast Retransmit
 - Fast Recovery
 - Big Window & Protection Against Wrapped Sequence Numbers option (RFC 1323)
- TCP NewReno includes the following features:
 - Slow Start
 - Congestion Avoidance
 - Fast Retransmit
 - Fast Recovery with modifications
- TCP Reno includes the following features:
 - Slow Start
 - Congestion Avoidance

- Fast Retransmit
- Fast Recovery
- TCP Selective ACKnowledgement (TCP SACK) includes the following features:
 - Slow Start
 - Congestion Avoidance
 - Fast Retransmit
 - Fast Recovery
 - Selective Acknowledgement
- TCP Tahoe includes the following features:
 - Slow Start
 - Congestion Avoidance
 - Fast Retransmit

7.3.2.2 Omitted Features

None.

7.3.2.3 Assumptions and Limitations

None.

7.3.3 Command Line Configuration

To configure TCP, include the parameters listed in [Table 7-17](#) in the scenario configuration (.config) file. See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 7-17. TCP Parameters

| Parameter | Value | Description |
|--|--|---|
| TCP <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none"> • LITE • NEWRENO • RENO • SACK • TAHOE <i>Default:</i> LITE | Variant of TCP. |
| TCP-DELAY-ACKS <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none"> • YES • NO <i>Default:</i> YES | Specifies whether ACKs for data segment packets are delayed for a certain period of time to reduce the number of ACKs transmitted. Note: If this parameter is set to NO, ACKs are sent immediately. |
| TCP-DELAY-SHORT-PACKETS-ACKS <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none"> • YES • NO <i>Default:</i> NO | Specifies whether acknowledgement of short packets are delayed. |

TABLE 7-17. TCP Parameters (Continued)

| Parameter | Value | Description |
|--|---|---|
| TCP-USE-NAGLE-ALGORITHM <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> YES | Specifies whether Nagle algorithm is implemented. In Nagle's algorithm, a source of TCP traffic will "coalesce" small outgoing data packets into fewer and larger outgoing packets. Note: In order to send small outgoing packets individually, disable this algorithm by changing this option to NO. |
| TCP-USE-RFC1323 <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Specifies whether the RFC 1323 option is enabled. The RFC 1323 option sends window scale and timestamps in the TCP options header. When enabled, the RFC 1323 option allows the reported window size to reach a maximum 1,073,725,440 bytes. Note: This option cannot be selected for TCP Reno or TCP Tahoe. |
| TCP-USE-KEEPALIVE-PROBES <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> YES | Specifies whether keep-alive probes are employed by TCP. To turn off these probes, change this option to NO. |
| TCP-MSS <i>Optional</i> Scope: Global, Node | Integer <i>Range:</i> ≥ 64 <i>Default:</i> 512 <i>Unit:</i> bytes | Specifies the maximum segment size (in bytes). |
| TCP-SEND-BUFFER <i>Optional</i> Scope: Global, Node | Integer <i>Range:</i> [64, 65535] <i>Default:</i> 16384 <i>Unit:</i> bytes | Specifies the send buffer size (in bytes). The send buffer size provides an upper bound on the advertised window. Note: Without RFC 1323, the maximum advertised window size is 65535. Setting this parameter to a value greater than 65535 will result in wasted memory without changing the behavior. Note: TCP-SEND-BUFFER should not be less than MSS. Since minimum value of MSS is 64, so it should not be less than 64. |
| TCP-RECEIVE-BUFFER <i>Optional</i> Scope: Global, Node | Integer <i>Range:</i> [1, 65535] or [1, 1073725440] <i>Default:</i> 16384 <i>Unit:</i> bytes | Specifies the receive buffer size (in bytes). The receive buffer size provide an upper bound on the advertised window. Note: The TCP sender cannot send more data than the available space of the receive buffer at the receiver. Note: With RFC 1323, the maximum receive buffer size can be 1073725440. |

TABLE 7-17. TCP Parameters (Continued)

| Parameter | Value | Description |
|---|---|--|
| TCP-USE-PUSH <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> YES | Specifies whether the push field is set by the TCP traffic source. Setting the push field results in immediate ACK for a given packet, except for FIN segments. By setting this option to NO, the push field is not set. |
| TCP-TRACE <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• NONE• TCPDUMP• TCPDUMP-ASCII <i>Default:</i> NONE | Specifies the type of TCP dump output generated. NONE : No TCP dump output is generated. TCPDUMP : TCP dump output is generated in the TCP Dump format (binary). The name of the dump file is tcptrace.dmp. TCPDUMP-ASCII : TCP dump output is generated in the TCP Dump format (ASCII). The name of the dump file is tcptrace.asc. The format of the dump file is described in Section 7.3.3.1 . |
| TCP-TRACE-DIRECTION <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• BOTH• INPUT• OUTPUT <i>Default:</i> BOTH | Specifies the type(s) of packets that are included in the dump file. INPUT : Only packets received from the network are recorded in the dump file. OUTPUT : Only packets sent to the network are recorded in the dump file. BOTH : Both sent and received packets are recorded in the dump file. |
| TCP-RANDOM-DROP-PERCENT <i>Optional</i> Scope: Global, Node | Real <i>Range:</i> [0.0, 100.0] <i>Default:</i> 0.0 | Specifies the initial random drop percent for the node. |
| TCP-VERIFICATION-DROP-COUNT <i>Optional</i> Scope: Global, Node | Integer <i>Range:</i> [0, 4] <i>Default:</i> 0 | Specifies the drop count. This option is for use with TCP verification scenarios that test variant behavior. Note: This parameter is for testing purposes. It is recommended that the default value be used for this parameter. |
| TCP-USE-OPTIONS <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Specifies whether TCP options are enabled or not. |

TABLE 7-17. TCP Parameters (Continued)

| Parameter | Value | Description |
|--|---|---|
| ECN <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Enables Explicit Congestion Notification (ECN). ECN marks the IP header, instead of dropping packets, when network queues report congestion. ECN requires IP-QUEUE-TYPE to be Random Early Detection (RED), RED for Input Output (RIO), or Weighted Red (WRED). Furthermore, the source and destination nodes must be ECN enabled; intermediate routes do not have to be ECN enabled. |
| TCP-STATISTICS <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Enables or disables TCP statistics collection. |
| TRACE-TCP <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Enables or disables the recording of the TCP header in the packet trace. Note: To enable packet tracing, some other parameters need to be configured as well. Refer to Section 4.2.10 of <i>QualNet User's Guide</i> for details. |

7.3.3.1 Format of the TCP Dump (ASCII) File

If the parameter TCP-TRACE is set to TCPDUMP-ASCII, then a dump file (with the name tcptrace.asc) is produced in the TCP Dump (ASCII) format.

Each line in this file has the following format:

```
<time> <src> > < dest>:<flags> <data-seqno> ack<ack seq no> win <value>
urg <options>
```

where

- | | |
|--------------|---|
| <time> | time is recorded in hour:minute:second.fraction |
| <src> | Source address. This can be either a host IP address or a network IP address. |
| <dest> | Destination address. This can be either a host IP address or a network IP address. |
| <flags> | flags are some combination of S (SYN), F (FIN), P (PUSH) or R (RST) or a single `.' (no flags). ACK is not considered a flag Multiple flags are concatenated together, e.g. FP to indicate FIN and PUSH |
| <data-seqno> | data-seqno describes the portion of sequence space covered by the data in this packet, i.e. <"initial sequence number of the packet": "initial sequence number of the packet + packet length">(packet length)> |

| | |
|------------------|---|
| ack <ack seq no> | ack is sequence number of the next data expected at the other direction on this connection. |
| win <value> | win is the number of bytes of "receive buffer space" available to the other direction on this connection. |
| urg | urg indicates there is 'urgent' data in the packet. |
| <options> | options are tcp options enclosed in angle brackets. Options are: <ul style="list-style-type: none">• nop for no operation• mss for mean segment size• wscale for window scale• timestamp for time stamp• sackOK for sack permitted• sack for sack options Multiple options are comma separated in single brackets. |

Notes: 1. All the above entries are on the same line.

2. <time>, <src>, <dst> and <flags> are always present.
3. The other fields depend on the contents of the packet's tcp protocol header and are output only if appropriate. For example, if the ack flag is missing, the ack field is not printed.

Examples:

The following are examples of TCPDUMP file (ASCII format):

```
0:0:10.000000000 0.0.1.1.1024 > 0.0.2.2.25: S 1:1(0) win 16384 <mss
512,sackOK,nop,nop>
```

```
0:0:2.440433600 0.0.1.1.1024 > 0.0.3.2.25: P 54002:55002(1000) ack
256002 win 28000
```

Note: Refer to the TCP Trace website at <http://www.tcptrace.com> for additional examples of the format.

7.3.4 GUI Configuration

This section describes how to configure TCP parameters in the GUI.

Configuring General TCP Parameters

To configure the general TCP parameters, perform the following steps:

1. Go to **Default Device Properties Editor > Node Configuration > Transport Layer**.
2. Set **Configure TCP** to Yes and set the dependent parameters listed in [Table 7-18](#).

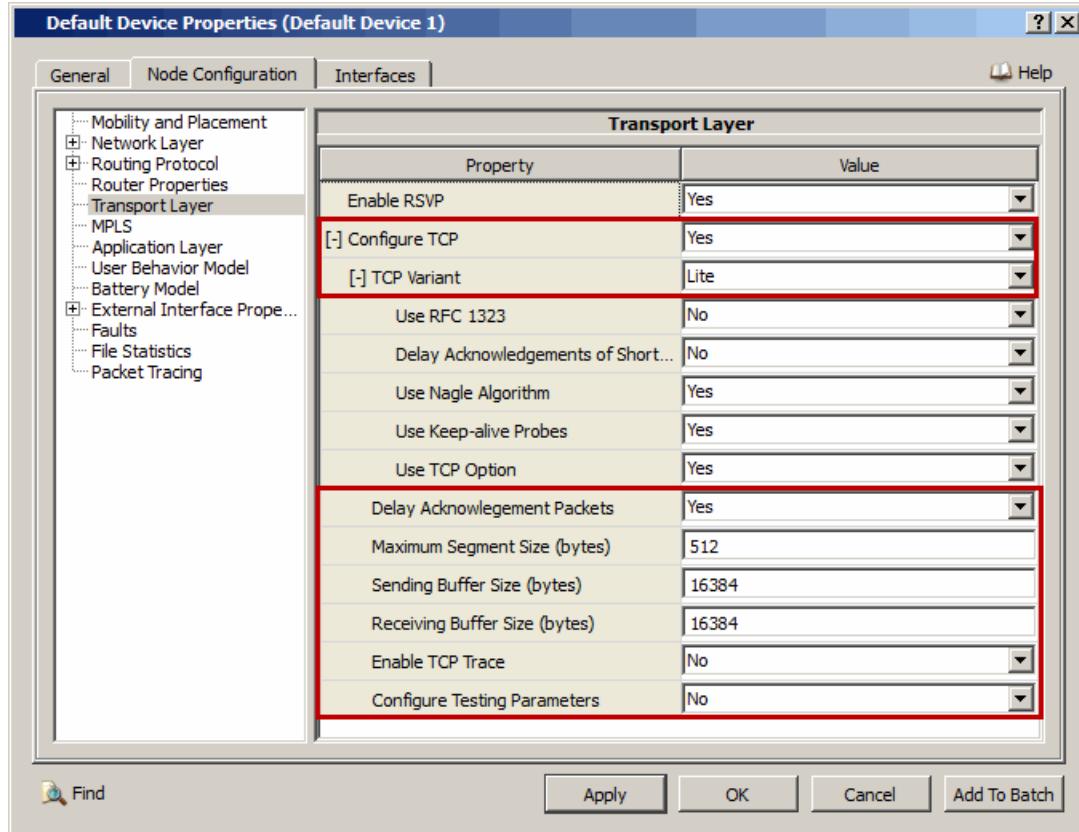


FIGURE 7-9. Configuring General TCP Parameters

TABLE 7-18. Command Line Equivalent of General TCP Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-------------------------------|------------------------|------------------------|
| TCP Variant | Node | TCP |
| Delay Acknowledgments Packets | Node | TCP-DELAY-ACKS |
| Maximum Segment Size | Node | TCP-MSS |
| Sending Buffer Size | Node | TCP-SEND-BUFFER |
| Receiving Buffer Size | Node | TCP-RECEIVE-BUFFER |
| Enable TCP Trace | Node | N/A |
| Configure Testing Parameters | Node | N/A |

Setting Parameters

- Set **Trace Variant** to the desired variant.
- To enable TCP trace, set **Enable TCP Trace** to Yes; otherwise, set **Enable TCP Trace** to No.
- To configure testing parameters, set **Configure Testing Parameters** to Yes; otherwise, set **Configure Testing Parameters** to No.

3. If **TCP Variant** is set to *Lite*, then set the dependent parameters listed in [Table 7-19](#).

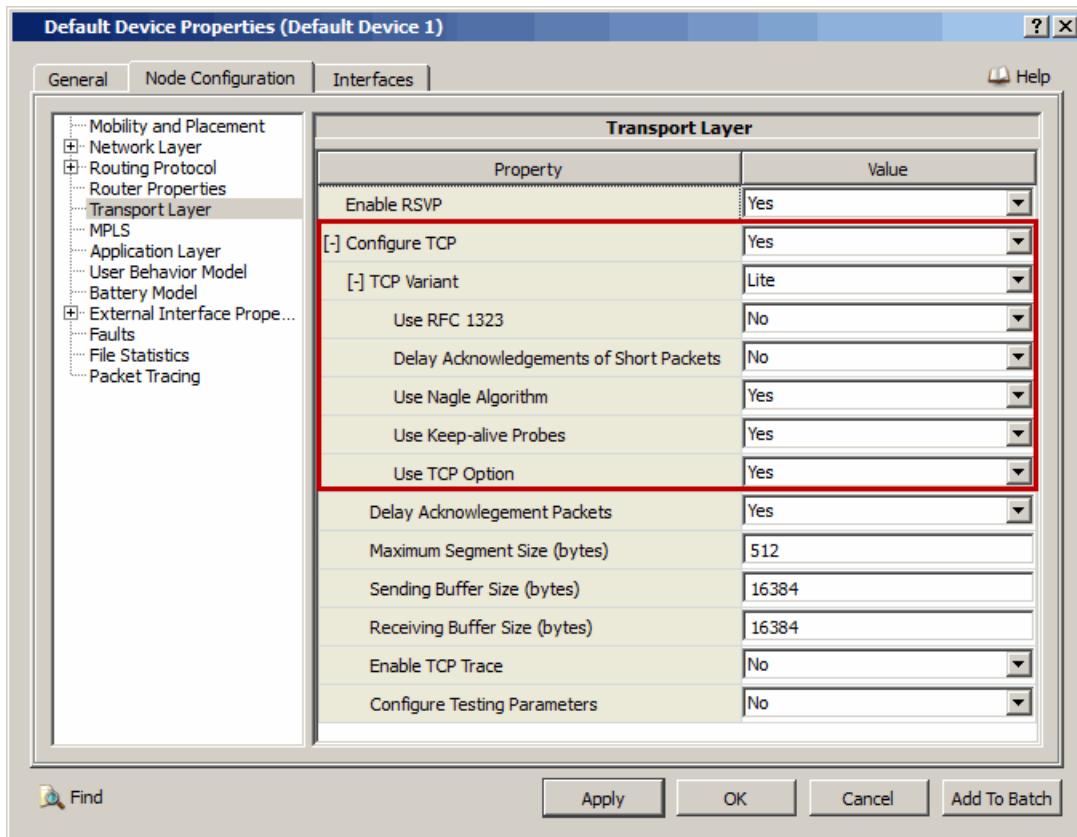


FIGURE 7-10. Setting TCP Lite Parameters

TABLE 7-19. Command Line Equivalent of TCP Lite Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|--|------------------------|------------------------------|
| Use RFC 1323 | Node | TCP-USE-RFC1323 |
| Delay Acknowledgments of Short Packets | Node | TCP-DELAY-SHORT-PACKETS-ACKS |
| Use Nagle Algorithm | Node | TCP-USE-NAGLE-ALGORITHM |
| Use Keep-alive Probes | Node | TCP-USE-KEEPALIVE-PROBES |
| Use TCP Option | Node | TCP-USE-OPTIONS |

4. If TCP Variant is set to *New Reno*, then set the dependent parameters listed in [Table 7-20](#).

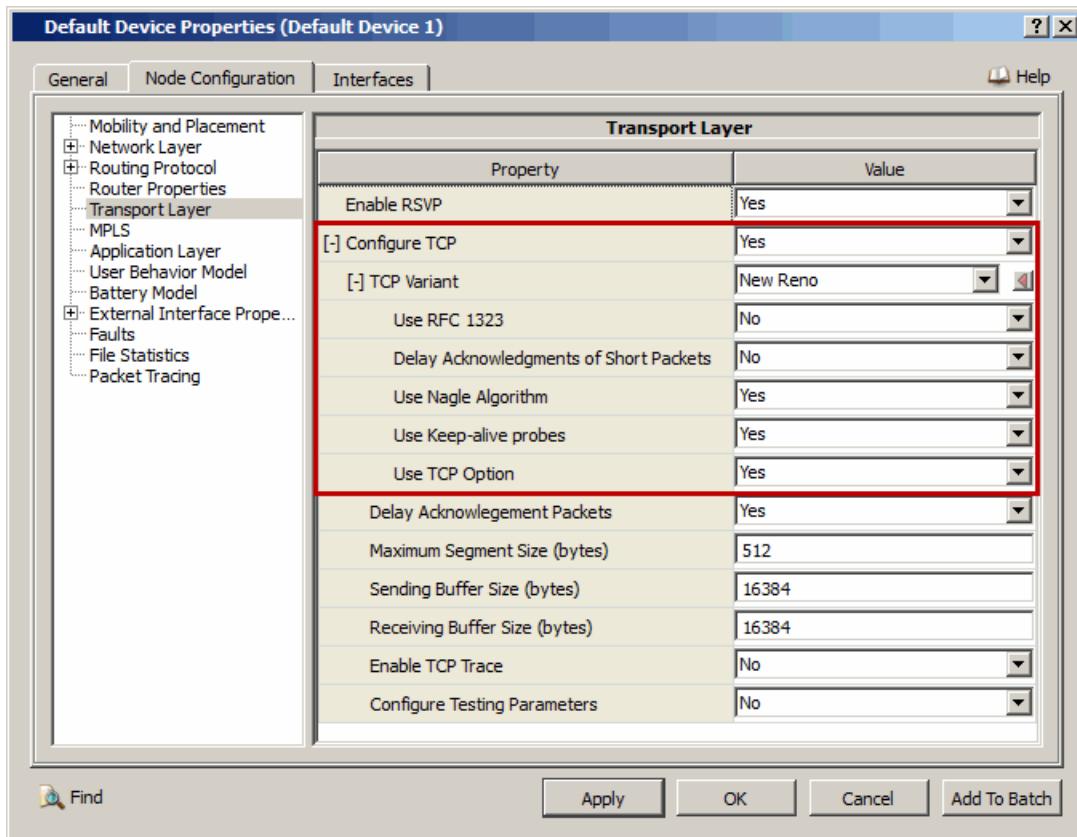


FIGURE 7-11. Setting TCP New Reno Parameters

TABLE 7-20. Command Line Equivalent of TCP New Reno Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|--|------------------------|------------------------------|
| Use RFC 1323 | Node | TCP-USE-RFC1323 |
| Delay Acknowledgments of Short Packets | Node | TCP-DELAY-SHORT-PACKETS-ACKS |
| Use Nagle Algorithm | Node | TCP-USE-NAGLE-ALGORITHM |
| Use Keep-alive Probes | Node | TCP-USE-KEEPALIVE-PROBES |
| Use TCP Option | Node | TCP-USE-OPTIONS |

5. If TCP Variant is set to *Reno*, then set the dependent parameters listed in Table 7-19.

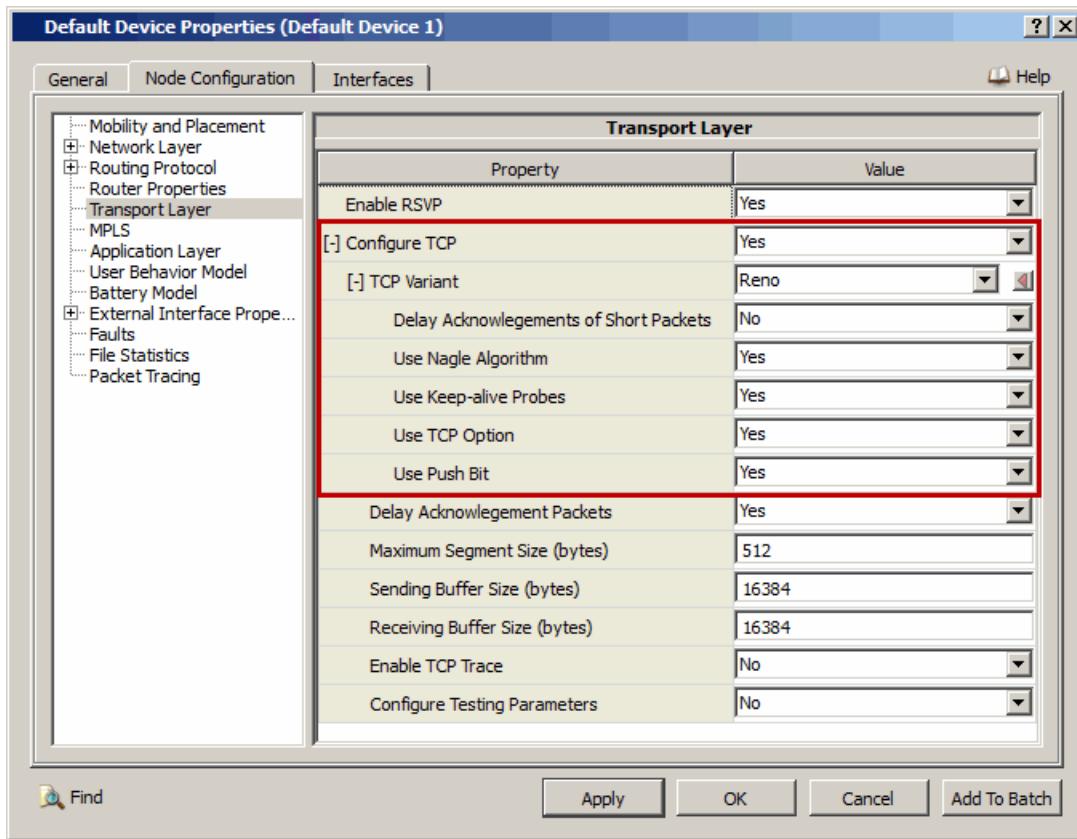


FIGURE 7-12. Setting TCP Reno Parameters

TABLE 7-21. Command Line Equivalent of TCP Reno Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|--|------------------------|------------------------------|
| Delay Acknowledgments of Short Packets | Node | TCP-DELAY-SHORT-PACKETS-ACKS |
| Use Nagle Algorithm | Node | TCP-USE-NAGLE-ALGORITHM |
| Use Keep-alive Probes | Node | TCP-USE-KEEPALIVE-PROBES |
| Use TCP Option | Node | TCP-USE-OPTIONS |
| Use Push Bit | Node | TCP-USE-PUSH |

6. If TCP Variant is set to SACK, then set the dependent parameters listed in Table 7-22.

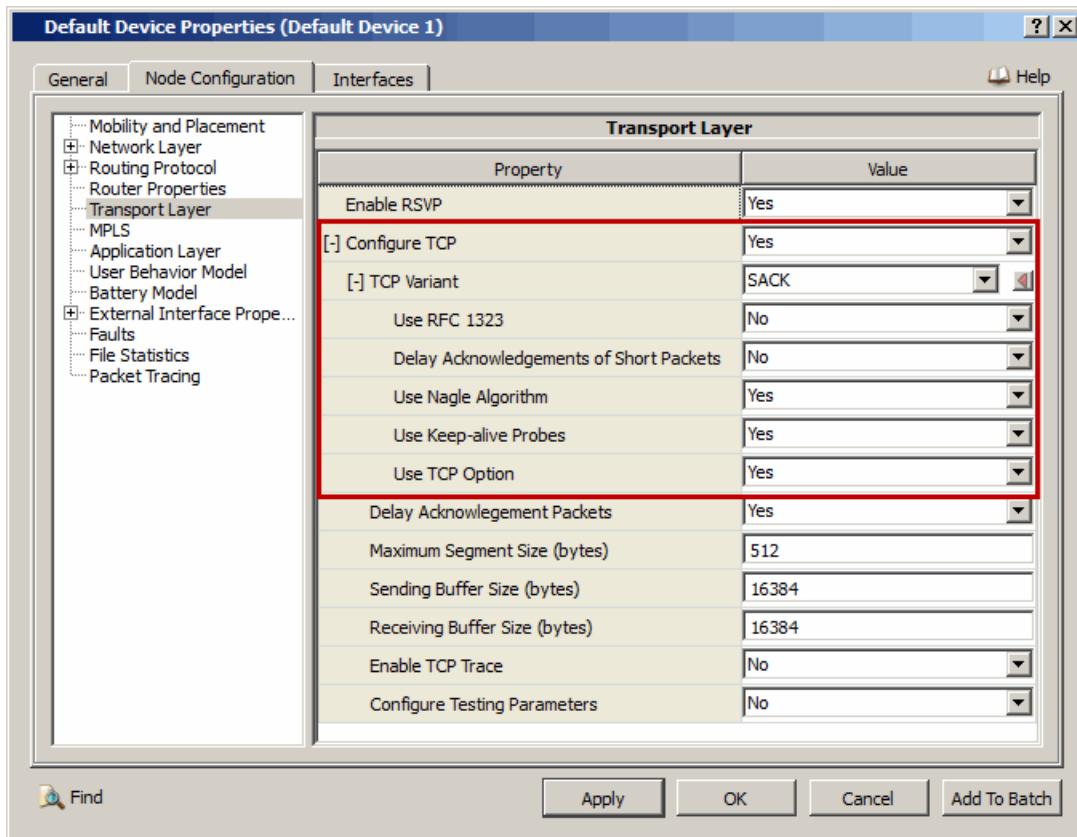


FIGURE 7-13. Setting TCP SACK Parameters

TABLE 7-22. Command Line Equivalent of TCP SACK Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|--|------------------------|------------------------------|
| Use RFC 1323 | Node | TCP-USE-RFC1323 |
| Delay Acknowledgments of Short Packets | Node | TCP-DELAY-SHORT-PACKETS-ACKS |
| Use Nagle Algorithm | Node | TCP-USE-NAGLE-ALGORITHM |
| Use Keep-alive Probes | Node | TCP-USE-KEEPALIVE-PROBES |
| Use TCP Option | Node | TCP-USE-OPTIONS |

7. If TCP Variant is set to *Tahoe*, then set the dependent parameters listed in Table 7-23.

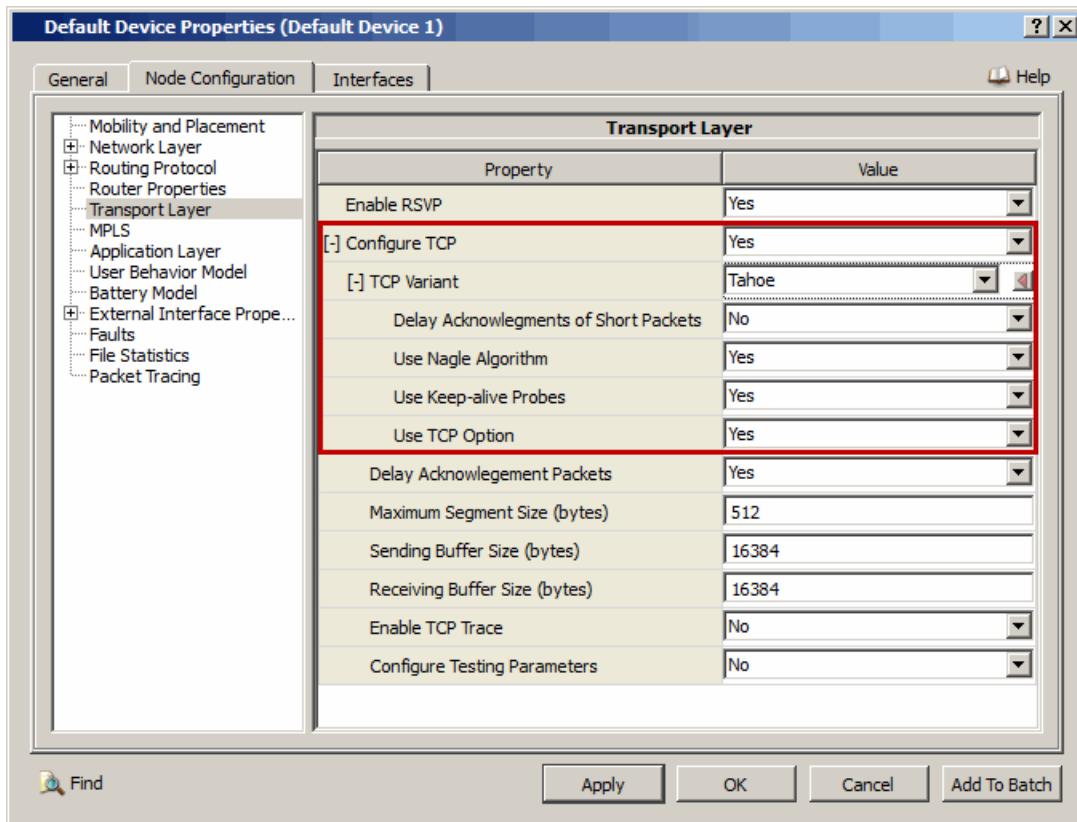


FIGURE 7-14. Setting TCP Tahoe Parameters

TABLE 7-23. Command Line Equivalent of TCP Tahoe Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|--|------------------------|------------------------------|
| Delay Acknowledgments of Short Packets | Node | TCP-DELAY-SHORT-PACKETS-ACKS |
| Use Nagle Algorithm | Node | TCP-USE-NAGLE-ALGORITHM |
| Use Keep-alive Probes | Node | TCP-USE-KEEPALIVE-PROBES |
| Use TCP Option | Node | TCP-USE-OPTIONS |

8. If **Configure TCP [= Yes] > Enable TCP Trace** is set to Yes, then set the dependent parameters listed in Table 7-24.

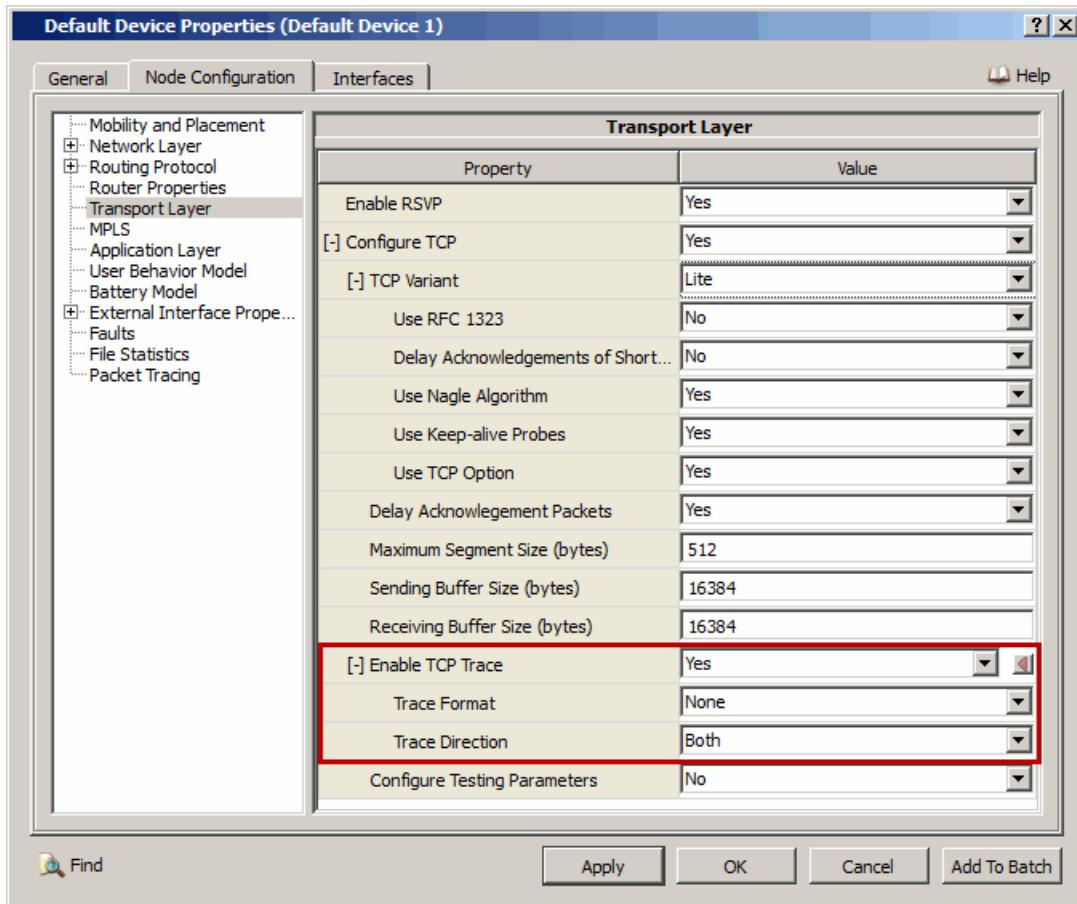


FIGURE 7-15. Setting TCP Trace Parameters

TABLE 7-24. Command Line Equivalent TCP Trace Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-----------------|------------------------|------------------------|
| Trace Format | Node | TCP-TRACE |
| Trace Direction | Node | TCP-TRACE-DIRECTION |

9. If **Configure TCP [= Yes] > Configure Testing Parameters** is set to Yes, then set the dependent parameters listed in [Table 7-25](#).

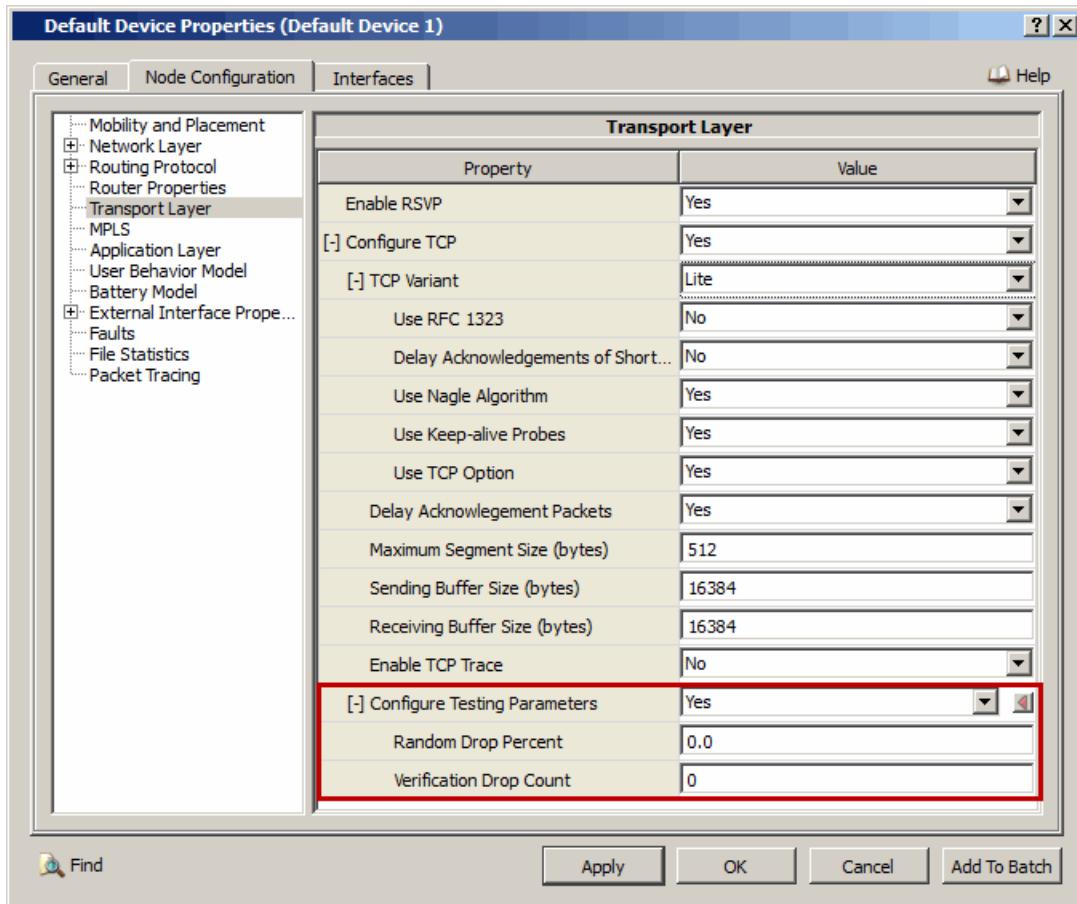


FIGURE 7-16. Setting TCP Testing Parameters

TABLE 7-25. Command Line Equivalent TCP Testing Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-------------------------|------------------------|-----------------------------|
| Random Drop Percent | Node | TCP-RANDOM-DROP-PERCENT |
| Verification Drop Count | Node | TCP-VERIFICATION-DROP-COUNT |

Configuring Statistics Parameters

Statistics for TCP can be collected at the global and node levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for Abstract TCP, check the box labeled **TCP** in the appropriate properties editor.

TABLE 7-26. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| TCP | Global, Node | TCP-STATISTICS |

Configuring Packet Tracing Parameters

Packet tracing for TCP can be enabled at the global and node levels. To enable packet tracing for TCP, in addition to setting the TCP trace parameter, **Trace TCP**, several other trace parameters also need to be set. See Section 4.2.10 of *QuaNet User's Guide* for details of configuring packet tracing parameters.

TABLE 7-27. Command Line Equivalent of Packet Tracing Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| Trace TCP | Global, Node | TRACE-TCP |

7.3.5 Statistics

This section describes the file, database, and dynamic statistics of the TCP model.

7.3.5.1 File Statistics

[Table 7-28](#) lists the TCP queue statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 7-28. TCP Statistics

| Statistic | Description |
|---|---|
| Packets Sent to Network Layer | Total number of packets sent to the network layer. |
| Data Packets Sent | Total number of packets sent including retransmitted packets. |
| Data Packets in Sequence | Total number of packets sent, excluding retransmissions and probes. |
| Data Packets Retransmitted | Total number of retransmitted packets. |
| Data Packets Fast Retransmitted | Total number of fast retransmitted packets. |
| ACK-only Packets Sent | Total number of acknowledgement-only packets sent. Piggyback acknowledgements are not sent. |
| Window Probes Sent | Total number of window probes sent. |
| Window Update-Only Packets Sent | Total number of window update packets sent. |
| Pure Control (SYN FIN RST) Packets Sent | Total number of pure control (SYN FIN RST) packets sent. |
| Total Packets Received From Network Layer | Total number of packets received from network layer. |
| Data Packets Received | Total number of data packets received. |
| Pure Control (SYN FIN RST) Packets Received | Total number of pure control (SYN FIN RST) packets received. |
| Duplicate ACK Packets Received | Total number of duplicate acknowledgement packets received. |

TABLE 7-28. TCP Statistics (Continued)

| Statistic | Description |
|---|--|
| In Sequence ACK Packets Received | Total number of In sequence acknowledgement packets received. |
| Window Probes Received | Total number of window probe packets received. |
| Window Update-Only Packets Received | Total number of window update packets received. |
| Total Packets with Errors | Total number of packets received with errors like checksum, offset, or short errors. |
| Packets Received with Checksum Errors | Total number of packets received with checksum errors. |
| Packets Received with Bad Offset | Total number of packets received with bad offset. |
| Packets Received that are Too Short | Total number of short packets received with packet size smaller than combined size of TCP and IP (without options) header. |
| Unicast data segments sent from the transport layer (segments) | Total number of unicast data segments sent from the transport layer. |
| Unicast data segments received at the transport layer (segments) | Total number of unicast data segments received at the transport layer. |
| Unicast data bytes sent from the transport layer (bytes) | Total number of unicast data bytes sent from the transport layer. |
| Unicast data bytes received at the transport layer (bytes) | Total number of unicast data bytes received at the transport layer. |
| Unicast overhead bytes sent from the transport layer (bytes) | Total number of unicast overhead bytes sent from the transport layer. |
| Unicast overhead bytes received at the transport layer (bytes) | Total number of unicast overhead bytes received at the transport layer. |
| Unicast control segments sent from the transport layer (segments) | Total number of unicast control segments sent from the transport layer. |
| Unicast control segments received at the transport layer (segments) | Total number of unicast control segments received at the transport layer. |
| Unicast control bytes sent from the transport layer (bytes) | Total number of unicast control bytes sent from the transport layer. |
| Unicast control bytes received at the transport layer (bytes) | Total number of unicast control bytes received at the transport layer. |
| Unicast offered load at the transport layer (bits/second) | Unicast offered load at the transport layer. |
| Unicast throughput at the transport layer (bits/second) | Unicast throughput at the transport layer. |
| Unicast goodput at the transport layer (bits/second) | Unicast goodput at the transport layer. |
| Unicast average delay at the transport layer (seconds) | Unicast average delay at the transport layer. |
| Unicast average delivery delay at the transport layer (seconds) | Unicast average delivery delay at the transport layer. |
| Unicast average jitter at the transport layer (seconds) | Unicast average jitter at the transport layer. |
| Unicast average delivery jitter at the transport layer (seconds) | Unicast average delivery jitter at the transport layer. |

7.3.5.2 Database Statistics

In addition to the file statistics, the TCP model also enters statistics in various scenario statistics database tables. Refer to *QualNet Statistics Database User's Guide* for details.

7.3.5.3 Dynamic Statistics

No dynamic statistics are supported for the TCP model.

7.3.6 Scenarios Included in QualNet

The QualNet distribution includes several sample scenarios for the TCP model. All scenarios are located in the directory `QUALNET_HOME/scenarios/developer/tcp`. [Table 7-29](#) lists the sub-directory where each scenario is located.

TABLE 7-29. TCP Scenarios Included in QualNet

| Scenario | Description |
|----------------|--|
| bottleneck-TCP | Shows multiple TCP flows share the same bottleneck link. |
| lite-0drop | Shows the behavior of the Lite in the presence of 0 packet drop. |
| lite-1drop | Shows the behavior of the Lite in the presence of 1 packet drop. |
| lite-2drop | Shows the behavior of the Lite in the presence of 2 packet drops. |
| lite-3drop | Shows the behavior of the Lite in the presence of 3 packet drops. |
| lite-4drop | Shows the behavior of the Lite in the presence of 4 packet drops. |
| newreno-0drop | Shows the behavior of the Newreno in the presence of 0 packet drop. |
| newreno-1drop | Shows the behavior of the Newreno in the presence of 1 packet drop. |
| newreno-2drop | Shows the behavior of the Newreno in the presence of 2 packet drops. |
| newreno-3drop | Shows the behavior of the Newreno in the presence of 3 packet drops. |
| newreno-4drop | Shows the behavior of the Newreno in the presence of 4 packet drops. |
| reno-0drop | Shows the behavior of the Reno in the presence of 0 packet drop. |
| reno-1drop | Shows the behavior of the Reno in the presence of 1 packet drop. |
| reno-2drop | Shows the behavior of the Reno in the presence of 2 packet drops |
| reno-3drop | Shows the behavior of the Reno in the presence of 3 packet drops. |
| reno-4drop | Shows the behavior of the Reno in the presence of 4 packet drops. |
| sack-0drop | Shows the behavior of the Sack in the presence of 0 packet drop. |
| sack-1drop | Shows the behavior of the Sack in the presence of 1 packet drop. |
| sack-2drop | Shows the behavior of the Sack in the presence of 2 packet drops. |
| sack-3drop | Shows the behavior of the Sack in the presence of 3 packet drops. |
| sack-4drop | Shows the behavior of the Sack in the presence of 4 packet drops. |
| tahoe-0drop | Shows the behavior of the Tahoe in the presence of 0 packet drop. |
| tahoe-1drop | Shows the behavior of the Tahoe in the presence of 1 packet drop. |
| tahoe-2drop | Shows the behavior of the Tahoe in the presence of 2 packet drops. |
| tahoe-3drop | Shows the behavior of the Tahoe in the presence of 3 packet drops. |
| tahoe-4drop | Shows the behavior of the Tahoe in the presence of 4 packet drops. |

Note: The network protocol used in all these scenarios is IPv4.

7.3.7 References

1. RFC 793, “Transmission Control Protocol”, Information Sciences Institute, University of Southern California. September 1981

2. RFC 879, "The TCP Maximum Segment Size and Related Topics", J. Postel. November 1983
3. RFC 896, "Congestion Control in IP/TCP Internetworks", John Nagle. January 1984
4. RFC 1122, "Requirements for Internet Hosts -- Communication Layers", R. Braden. October 1989
5. TCP/IP Illustrated, Vol 1 and 2 – The Protocols. W Richard Stevens, Addison Wesley
6. RFC 2481, "A Proposal to add Explicit Congestion Notification (ECN) to IP", S. Floyd. January 1999
7. RFC 2884, "Performance Evaluation of Explicit Congestion Notification (ECN) in IP Networks", J. Hadi Salim, U. Ahmed. July 2000
8. RFC 2309, "Recommendations on Queue Management and Congestion Avoidance in the Internet", B. Braden, D. Clark, J. Crowcroft, B. Davie, S. Deering, D. Estrin, S. Floyd, V. Jacobson, G. Minshall, C. Partridge, L. Peterson, K. Ramakrishnan, S. Shenkar, J. Wroclawski, L. Zhand. April 2998
9. RFC 791, "Internet Protocol Darpa Internet Program Protocol Specification", Information Sciences Institute, University of Southern California. September 1981
- 10.RFC 2582, "The Newreno Modification to TCP's Fast Recovery Algorithm", S. Floyd, T. Henderson. April 1999.
- 11.RFC 2001, "TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery", W. Stevens. January 1997.
- 12.RFC 2018, "TCP Selective Acknowledgment Options", M. Mathis, J. Mahdavi, S. Floyd, A. Romanow. October 1996.
- 13.RFC 1323, "TCP Extensions for High Performance", V. Jacobson, R. Braden, D. Borman. May 1992.
- 14.<http://www.tcptrace.com/>
- 15.<http://www.tcpdump.org/>
- 16.<http://www.tac.eu.org/cgi-bin/man-cgi?tcpdump>
- 17.Issues of TCP with SACK, Sally Floyd, http://www.aciri.org/floyd/papers/issues_sacks.pdf
- 18.Simulation-based Comparisons of Tahoe, Reno, and SACK TCP, Kevin Fall and Sally Floyd, <http://www.aciri.org/floyd/papers/sacks.pdf>

7.4 User Datagram Protocol (UDP)

The QualNet UDP model is based on RFC 768.

7.4.1 Description

UDP is a simple, low-overhead, packet-switched transport protocol that assumes that the Internet Protocol (IP) is the underlying network protocol. Applications that require UDP include CBR or routing protocols such as RIP.

7.4.2 Command Line Configuration

The UDP model is always enabled and does not require to be enabled explicitly.

UDP Parameters

Table 7-30 shows the UDP parameters. See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 7-30. UDP Parameters

| Parameter | Value | Description |
|---|---|---|
| UDP-STATISTICS Optional Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Enables or disables collection of UDP statistics. |
| TRACE-UDP Optional Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Enables or disables the recording of the UDP header in the packet trace. Note: To enable packet tracing, some other parameters need to be configured as well. Refer to Section 4.2.10 of <i>QualNet User's Guide</i> for details. |

7.4.3 GUI Configuration

UDP is always enabled and does not require to be enabled explicitly. This section describes how to configure statistics and packet tracing parameters for UDP in the GUI.

Configuring Statistics Parameters

Statistics for UDP can be collected at the global and node levels. See [Section 4.2.9 of QualNet User's Guide](#) for details of configuring statistics parameters.

To enable statistics collection for UDP, check the box labeled **UDP** in the appropriate properties editor.

TABLE 7-31. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|----------------------|-------------------------------|-------------------------------|
| UDP | Global, Node | UDP-STATISTICS |

Configuring Packet Tracing Parameters

Packet tracing for UDP can be enabled at the global and node levels. To enable packet tracing for UDP, in addition to setting the UDP trace parameter, *Trace UDP*, several other trace parameters also need to be set. See Section 4.2.10 of *QuaNet User's Guide* for details of configuring packet tracing parameters.

TABLE 7-32. Command Line Equivalent of Packet Tracing Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|----------------------|-------------------------------|-------------------------------|
| Trace UDP | Global, Node | TRACE-UDP |

7.4.4 Statistics

This section describes the file, database, and dynamic statistics of the UDP model.

7.4.4.1 File Statistics

[Table 7-33](#) lists the UDP statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 7-33. UDP Statistics

| Statistic | Description |
|---|---|
| Unicast data segments sent from the transport layer (segments) | Total number of unicast data segments sent from the transport layer. |
| Unicast data segments received at the transport layer (segments) | Total number of unicast data segments received at the transport layer. |
| Unicast data bytes sent from the transport layer (bytes) | Total number of unicast data bytes sent from the transport layer. |
| Unicast data bytes received at the transport layer (bytes) | Total number of unicast data bytes received at the transport layer. |
| Unicast overhead bytes sent from the transport layer (bytes) | Total number of unicast overhead bytes sent from the transport layer. |
| Unicast overhead bytes received at the transport layer (bytes) | Total number of unicast overhead bytes received at the transport layer. |
| Unicast control segments sent from the transport layer (segments) | Total number of unicast control segments sent from the transport layer. |
| Unicast control segments received at the transport layer (segments) | Total number of unicast control segments received at the transport layer. |
| Unicast control bytes sent from the transport layer (bytes) | Total number of unicast control bytes sent from the transport layer. |

TABLE 7-33. UDP Statistics (Continued)

| Statistic | Description |
|---|---|
| Unicast control bytes received at the transport layer (bytes) | Total number of unicast control bytes received at the transport layer. |
| Unicast offered load at the transport layer (bits/second) | Unicast offered load at the transport layer. |
| Unicast throughput at the transport layer (bits/second) | Unicast throughput at the transport layer. |
| Unicast goodput at the transport layer (bits/second) | Unicast goodput at the transport layer. |
| Unicast average delay at the transport layer (seconds) | Unicast average delay at the transport layer. |
| Unicast average delivery delay at the transport layer (seconds) | Unicast average delivery delay at the transport layer. |
| Unicast average jitter at the transport layer (seconds) | Unicast average jitter at the transport layer. |
| Unicast average delivery jitter at the transport layer (seconds) | Unicast average delivery jitter at the transport layer. |
| Broadcast data segments sent from the transport layer (segments) | Total number of broadcast data segments sent from the transport layer. |
| Broadcast data segments received at the transport layer (segments) | Total number of broadcast data segments received at the transport layer. |
| Broadcast data bytes sent from the transport layer (bytes) | Total number of broadcast data bytes sent from the transport layer. |
| Broadcast data bytes received at the transport layer (bytes) | Total number of broadcast data bytes received at the transport layer. |
| Broadcast overhead bytes sent from the transport layer (bytes) | Total number of broadcast overhead bytes sent from the transport layer. |
| Broadcast overhead bytes received at the transport layer (bytes) | Total number of broadcast overhead bytes received at the transport layer. |
| Broadcast control segments sent from the transport layer (segments) | Total number of broadcast control segments sent from the transport layer. |
| Broadcast control segments received at the transport layer (segments) | Total number of broadcast control segments received at the transport layer. |
| Broadcast control bytes sent from the transport layer (bytes) | Total number of broadcast control bytes sent from the transport layer. |
| Broadcast control bytes received at the transport layer (bytes) | Total number of broadcast control bytes received at the transport layer. |
| Broadcast offered load at the transport layer (bits/second) | Broadcast offered load at the transport layer. |
| Broadcast throughput at the transport layer (bits/second) | Broadcast throughput at the transport layer. |
| Broadcast goodput at the transport layer (bits/second) | Broadcast goodput at the transport layer. |
| Broadcast average delay at the transport layer (seconds) | Broadcast average delay at the transport layer. |
| Broadcast average delivery delay at the transport layer (seconds) | Broadcast average delivery delay at the transport layer. |
| Broadcast average jitter at the transport layer (seconds) | Broadcast average jitter at the transport layer. |
| Broadcast average delivery jitter at the transport layer (seconds) | Broadcast average delivery jitter at the transport layer. |
| Multicast data segments sent from the transport layer (segments) | Total number of multicast data segments sent from the transport layer. |
| Multicast data segments received at the transport layer (segments) | Total number of multicast data segments received at the transport layer. |

TABLE 7-33. UDP Statistics (Continued)

| Statistic | Description |
|---|---|
| Multicast data bytes sent from the transport layer (bytes) | Total number of multicast data bytes sent from the transport layer. |
| Multicast data bytes received at the transport layer (bytes) | Total number of multicast data bytes received at the transport layer. |
| Multicast overhead bytes sent from the transport layer (bytes) | Total number of multicast overhead bytes sent from the transport layer. |
| Multicast overhead bytes received at the transport layer (bytes) | Total number of multicast overhead bytes received at the transport layer. |
| Multicast control segments sent from the transport layer (segments) | Total number of multicast control segments sent from the transport layer. |
| Multicast control segments received at the transport layer (segments) | Total number of multicast control segments received at the transport layer. |
| Multicast control bytes sent from the transport layer (bytes) | Total number of multicast control bytes sent from the transport layer. |
| Multicast control bytes received at the transport layer (bytes) | Total number of multicast control bytes received at the transport layer. |
| Multicast offered load at the transport layer (bits/second) | Multicast offered load at the transport layer. |
| Multicast throughput at the transport layer (bits/second) | Multicast throughput at the transport layer. |
| Multicast goodput at the transport layer (bits/second) | Multicast goodput at the transport layer. |
| Multicast average delay at the transport layer (seconds) | Multicast average delay at the transport layer. |
| Multicast average delivery delay at the transport layer (seconds) | Multicast average delivery delay at the transport layer. |
| Multicast average jitter at the transport layer (seconds) | Multicast average jitter at the transport layer. |
| Multicast average delivery jitter at the transport layer (seconds) | Multicast average delivery jitter at the transport layer. |

7.4.4.2 Database Statistics

In addition to the file statistics, the UDP model also enters statistics in various scenario statistics database tables. Refer to *QualNet Statistics Database User's Guide* for details.

7.4.4.3 Dynamic Statistics

No dynamic statistics are supported for UDP.

7.4.5 References

1. RFC 768, "User Datagram Protocol", J. Postel. August 1980.

8 Application Layer Models

This chapter describes features, configuration requirements and parameters, statistics, and scenarios for Application Layer Models, and consists of the following sections:

- Background Traffic Model
- Constant Bit Rate (CBR) Traffic Generator
- File Transfer Protocol (FTP)
- File Transfer Protocol/Generic (FTP/Generic)
- HyperText Transfer Protocol (HTTP)
- Lookup Traffic Generator
- Multicast Constant Bit Rate (MCBR) Traffic Generator
- Super Application Traffic Generator
- Telecommunications Network (TELNET)
- Traffic Generator (Traffic-Gen)
- Trace File-based Traffic Generator (Traffic-Trace)
- Variable Bit Rate (VBR) Traffic Generator

8.1 Background Traffic Model

8.1.1 Description

The Background Traffic model allows the user to model background traffic with specific priorities on wired and wireless point-to-point links. The background traffic is not modeled by actual transmission of traffic packets but modeled as a reduction in available bandwidth on the link.

8.1.2 Command Line Configuration

To specify background traffic in a scenario, include the following parameter in the scenario configuration (.config) file:

```
BACKGROUND-TRAFFIC-CONFIG-FILE <bgtraffic-file>
```

where

<bgtraffic-file> Name of the Background Traffic file.

The format of this file is described in [Section 8.1.2.1](#).

8.1.2.1 Format of the Background Traffic File

Each line in the Background Traffic file specifies a background traffic flow on one of the wired or wireless point-to-point links in the scenario. Each line in the file has the following format (all parameters should be entered on the same line):

```
<interface> LINK-BACKGROUND-TRAFFIC<flow-id> <start-time> <duration>
                                         <bandwidth> [<priority>]
```

where

<interface> IP address of the interface that is the source of the background traffic, enclosed in [and].

<flow-id> Flow ID of the background traffic.

This is an integer, starting from 0, enclosed in [and].

<start-time> Time when the background traffic flow starts.

The start time is specified as a time number distribution. See note below.

<duration> Duration of the background traffic flow.

The duration is specified as a time number distribution. See note below.

<bandwidth> Bandwidth consumed by the background traffic flow.

The bandwidth is specified as an integer distribution. See note below.

<priority>

Priority of the background traffic flow.

Priority can be specified by including a TOS specification, DSCP specification, or Precedence specification.

- TOS specification has the following format:

TOS <TOS-value>

where <TOS-value> is the value of the TOS bits of the IP header. <TOS-value> should be an integer in the range [0, 255].

- DSCP specification has the following format:

DSCP <DSCP-value>

where <DSCP-value> is the value of the DSCP bits of the IP header. <DSCP-value> should be an integer in the range [0, 63].

- Precedence specification has the following format:

PRECEDENCE <precedence-value>

where <precedence-value> is the value of the Precedence bits of the IP header. <precedence-value> should be an integer in the range [0, 7].

At most one of the three parameters TOS, DSCP, or PRECEDENCE can be specified. If the priority is not specified, PRECEDENCE 0 is used as default.

Notes: Integer and Time Distributions: Several parameters are specified as random number distributions. The following random number distributions are supported: deterministic, uniform, and exponential.

- The deterministic distribution is specified as:

```
DET <det-val>
```

It always returns `<det-val>` as the value.

- The uniform distribution is specified as:

```
UNI <uni-val-1> <uni-val-2>
```

It returns a value uniformly distributed between `<uni-val-1>` and `<uni-val-2>`.

- The exponential distribution is specified as:

```
EXP <exp-val>
```

It returns a value from an exponential distribution with `<exp-val>` as the mean.

For integer distributions, `<det-val>`, `<uni-val-1>`, `<uni-val-2>`, and `<exp-val>` are integer values, e.g., 0, 10, 15, etc.

For time distributions, `<det-val>`, `<uni-val-1>`, `<uni-val-2>`, and `<exp-val>` are time values, e.g., 5S, 0.5MS, 100US, etc.

Example of Background Traffic File

The following is an example of a Background Traffic file:

```
[190.0.1.1] LINK-BACKGROUND-TRAFFIC[0] DET 0S DET 100S DET 5000 TOS 100
[190.0.1.2] LINK-BACKGROUND-TRAFFIC[0] EXP 0S UNI 3S 6S DET 23000
[190.0.1.1] LINK-BACKGROUND-TRAFFIC[1] DET 0S DET 2M DET 25000
[190.0.2.1] LINK-BACKGROUND-TRAFFIC[0] DET 0S DET 34M DET 25000 DSCP 2
```

8.1.3 GUI Configuration

This section describes how to configure background traffic in the GUI.

To configure background traffic on a point-to-point link, perform the following steps:

1. Go to the **Point-to-point Link Properties Editor > Point-to-point Link Properties > Background Traffic**. This opens the Background Traffic editor in the right panel of the properties editor.

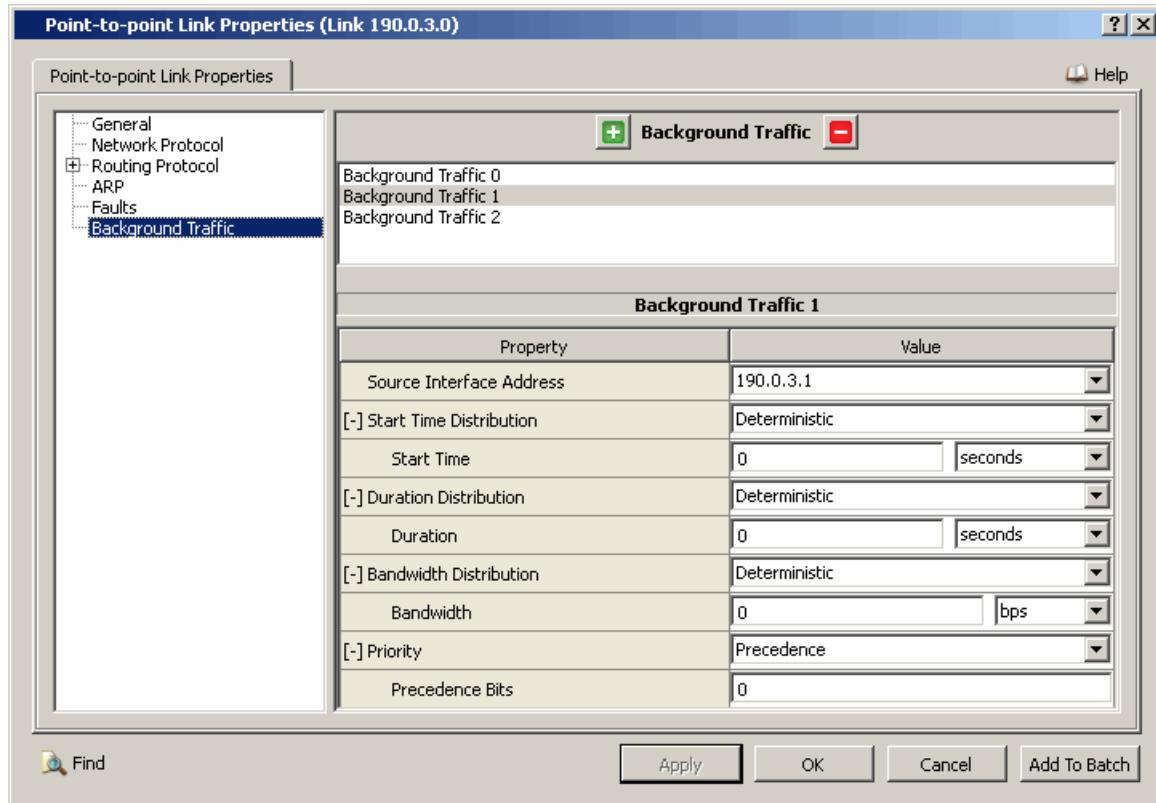


FIGURE 8-1. Background Traffic Editor

2. To create a new background traffic flow, click on the button. To delete a background traffic flow in the top panel of the editor, select the traffic flow in the top panel of the editor and click on the button. To configure the properties a traffic flow, select the traffic flow and set the properties in the bottom panel.

3. To specify the source of the traffic flow, select the source interface address from the pull-down list in the **Value** field of the parameter **Interface Address**.

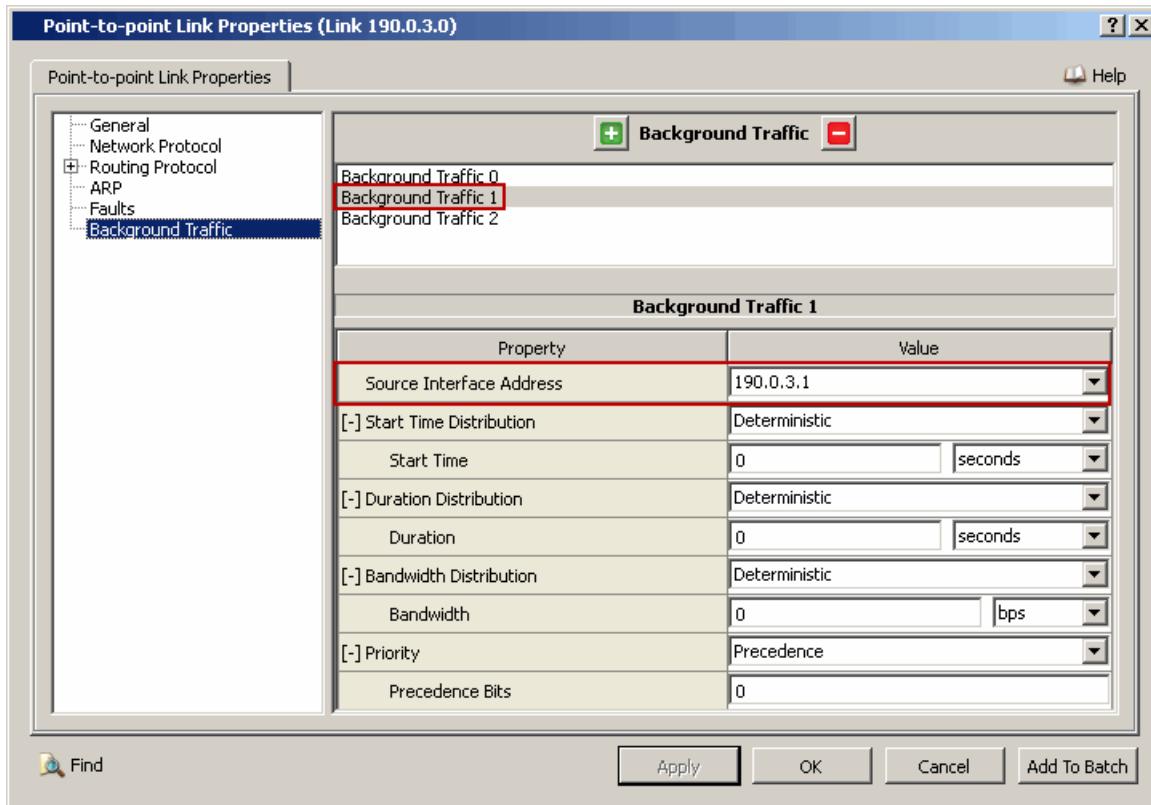


FIGURE 8-2. Specifying Source Address

TABLE 8-1. Command Line Equivalent of Source Address Parameters

| GUI Parameter | Command Line Parameter |
|--------------------------|------------------------|
| Source Interface Address | <interface> |

4. To specify the start time, set the parameter **Start Time Distribution** and set the dependent parameters of the selected distribution.
- If **Start Time Distribution** is set to *Deterministic*, then set the dependent parameters listed in [Table 8-2](#).

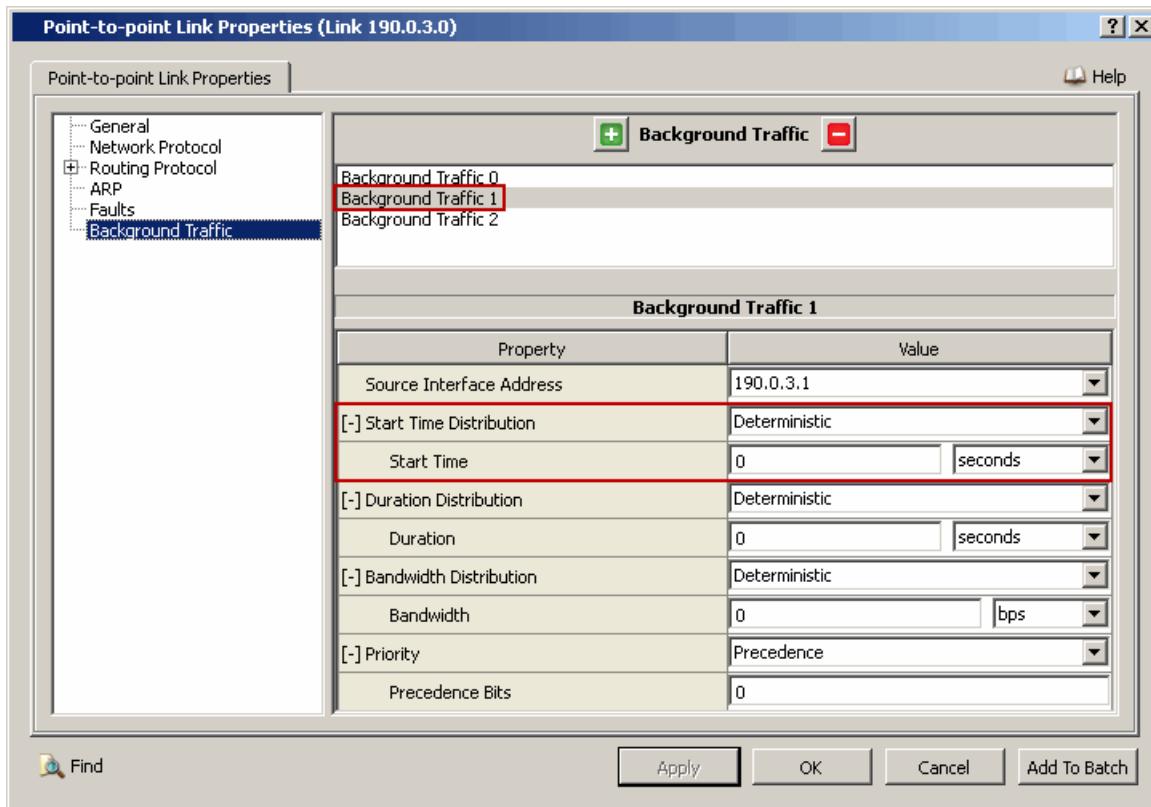


FIGURE 8-3. Setting Parameters for a Deterministic Distribution

TABLE 8-2. Command Line Equivalent of Deterministic Distribution Parameters

| GUI Parameter | Command Line Parameter |
|--|------------------------|
| Start Time Distribution (set to <i>Deterministic</i>) | DET |
| Start Time | <det-val> |

- If **Start Time Distribution** is set to *Exponential*, then set the dependent parameters listed in Table 8-3.

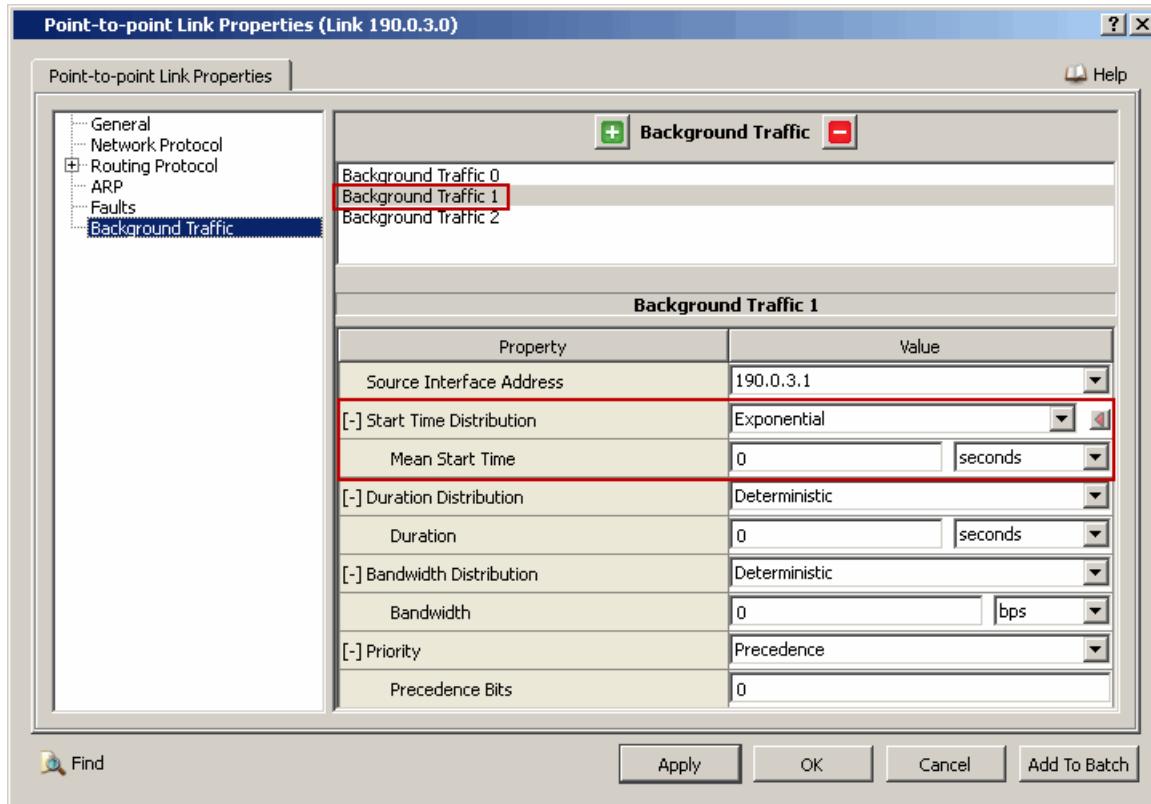


FIGURE 8-4. Setting Parameters for an Exponential Distribution

TABLE 8-3. Command Line Equivalent of Exponential Distribution Parameters

| GUI Parameter | Command Line Parameter |
|--|------------------------|
| Start Time Distribution (set to <i>Exponential</i>) | EXP |
| Mean Start Time | <exp-val> |

- If **Start Time Distribution** is set to *Uniform* then set the dependent parameters listed in [Table 8-4](#).

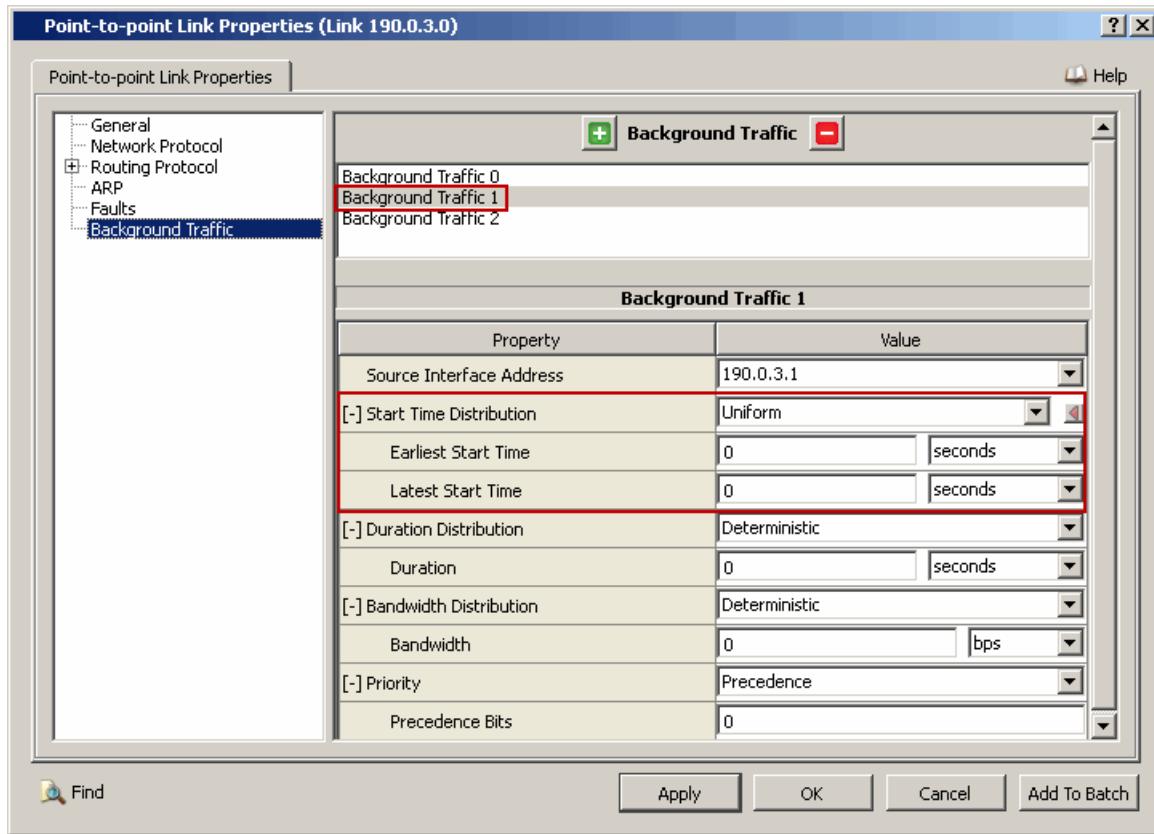


FIGURE 8-5. Setting Parameters for a Uniform Distribution

TABLE 8-4. Command Line Equivalent of Uniform Distribution Parameters

| GUI Parameter | Command Line Parameter |
|--|------------------------|
| Start Time Distribution (set to <i>Uniform</i>) | UNI |
| Earliest Start Time | <uni-val-1> |
| Latest Start Time | <uni-val-2> |

- To specify the duration, set the parameter **Duration Distribution** and set the dependent parameters of the selected distribution in the same way as for **Start Time Distribution**.
- To specify the bandwidth, set the parameter **Bandwidth Distribution** and set the dependent parameters of the selected distribution in the same way as for **Start Time Distribution**.

7. To specify a priority for the traffic flow, set **Priority** to *DSCP*, *Precedence*, or *TOS*. If **Priority** is set to *DSCP*, *Precedence*, or *TOS*, set the dependent parameters listed in Table 8-5, Table 8-6, and Table 8-7, respectively. Figure 8-6 shows how to set the dependent parameters when **Priority** is set to *DSCP*. Setting dependent parameters for the other two options is similar.

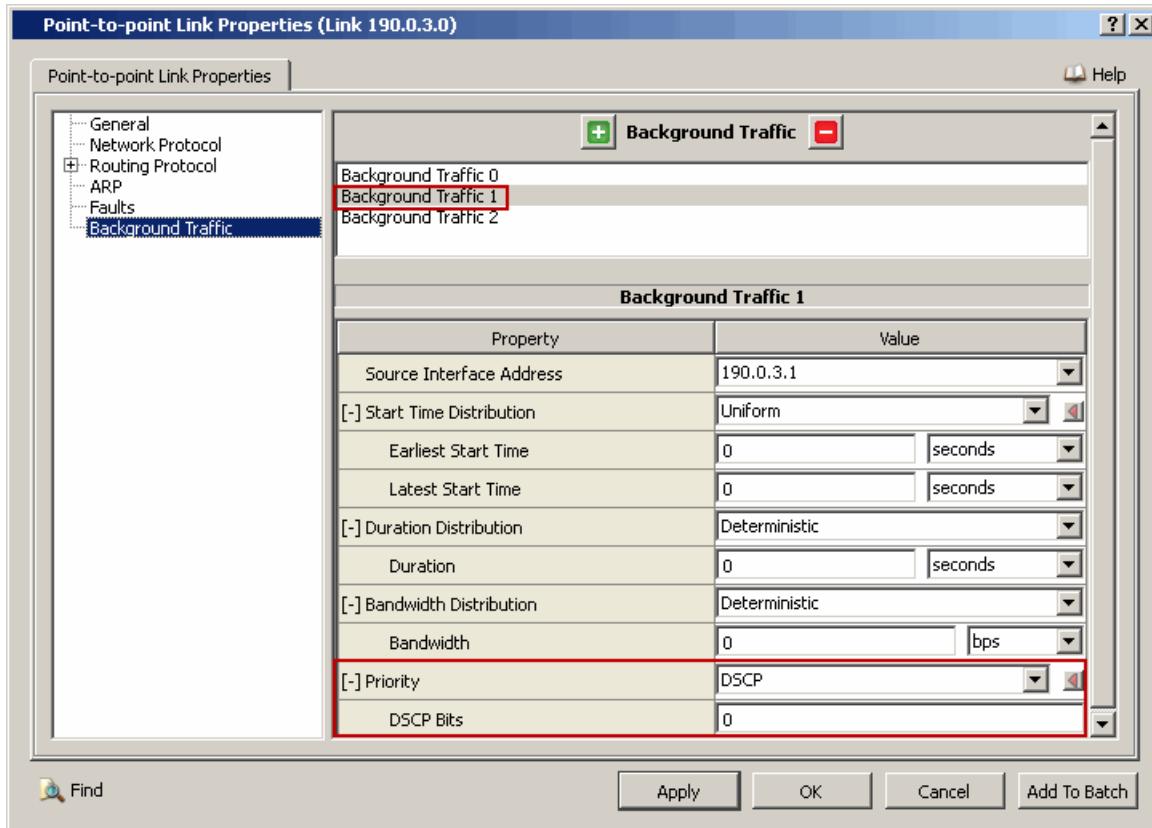


FIGURE 8-6. Setting DSCP Value

TABLE 8-5. Command Line Equivalent of Priority Parameters (Priority = DSCP)

| GUI Parameter | Command Line Parameter |
|--------------------------------|------------------------|
| Priority (set to <i>DSCP</i>) | DSCP |
| DSCP Value | <DSCP-value> |

TABLE 8-6. Command Line Equivalent of Priority Parameters (Priority = Precedence)

| GUI Parameter | Command Line Parameter |
|--------------------------------------|------------------------|
| Priority (set to <i>Precedence</i>) | PRECEDENCE |
| Precedence Value | <precedence-value> |

TABLE 8-7. Command Line Equivalent of Priority Parameters (Priority = TOS)

| GUI Parameter | Command Line Parameter |
|-------------------------------|------------------------|
| Priority (set to <i>TOS</i>) | TOS |
| TOS Value | <TOS-value> |

8.1.4 Statistics

No statistics are collected for the Background Traffic model.

8.1 Constant Bit Rate (CBR) Traffic Generator

8.1.1 Description

The Constant Bit Rate (CBR) traffic generator generates traffic at a constant rate by transmitting packets (also called “items”) of a fixed size at a fixed rate. It is generally used to provide background traffic that affects the performance of other applications being analyzed or to simulate generic multimedia traffic.

CBR can be used to simulate applications for which the end-systems require predictable response time and a static amount of bandwidth is continuously available for the life-time of the connection. These applications include services such as video-conferencing and telephony (voice services).

8.1.2 Features and Assumptions

This section describes the implemented features, omitted features, assumptions and limitations of the CBR model.

8.1.2.1 Implemented Features

- Transmission of packets of a fixed size with a constant inter-packet time.
- Option to specify QoS for data flows.
- Option to use RSVP-TE for MPLS networks.

8.1.2.2 Omitted Features

None.

8.1.2.3 Assumptions and Limitations

None.

8.1.3 Command Line Configuration

Application Configuration File Parameters

To specify CBR traffic, include the following statement in the application configuration (.app) file:

```
CBR    <src> <dest> <items-to-send> <item-size> <interval>
      <start-time> <end-time>
      [TOS <tos-value> | DSCP <dscp-value> |
       PRECEDENCE <precedence-value>]
      [RSVP-TE]
      [MDP-ENABLED [MDP-PROFILE <profile-name>] ]
      [APPLICATION-NAME <application-name>]
```

Note: All parameters should be entered on the same line.

The CBR parameters are described in [Table 8-1](#). See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 8-1. CBR Parameters

| Parameter | Value | Description |
|--|--|--|
| <src> <i>Required</i> | Integer or IP Address | Client node's ID or IP address. |
| <dest> <i>Required</i> | Integer or IP Address | Server node's ID or IP address. |
| <items-to-send> <i>Required</i> | Integer <i>Range:</i> ≥ 0 | Number of packets to send. If this is set to 0, items will be sent continually until <end-time> or until the end of the simulation, whichever comes first. See note 1. |
| <item-size> <i>Required</i> | Integer <i>Range:</i> [32, 65023] <i>Unit:</i> bytes | Size of each item. |
| <interval> <i>Required</i> | Time <i>Range:</i> $> 0\text{S}$ | Time between transmission of successive packets (inter-departure time). |
| <start-time> <i>Required</i> | Time <i>Range:</i> $\geq 0\text{S}$ | Time when the transmission of packets should begin. |
| <end-time> <i>Required</i> | Time <i>Range:</i> $\geq 0\text{S}$ | Time when the transmission of packets should end. If this is set to 0, transmission ends after <items-to-send> packets have been sent or until the end of simulation, whichever comes first. Note: <end-time> should be 0 or greater than <start-time>. |
| TOS <tos-value> <i>Optional</i> | Integer <i>Range:</i> [0, 255] | Value of the 8-bit TOS field of the IP header for the packets generated. See note 2. |
| DSCP <dscp-value> <i>Optional</i> | Integer <i>Range:</i> [0, 63] | Value of the 6-bit DSCP field of the IP header for the packets generated. See note 2. |
| PRECEDENCE <precedence-value> <i>Optional</i> | Integer <i>Range:</i> [0, 7] | Value of the 3-bit Precedence field of the IP header for the packets generated. See note 2. |
| RSVP-TE <i>Optional</i> | | Specifies that RSVP-TE is used to transmit packets over an MPLS network. |
| MDP-ENABLED <i>Optional</i> | N/A | Keyword which specifies that MDP is enabled for the application. Note: If this keyword is not included, then the application does not run with MDP. |

TABLE 8-1. CBR Parameters (Continued)

| Parameter | Value | Description |
|--|--------|---|
| MDP-PROFILE <i><profile-name></i> <i>Optional</i> | String | <p>Name of the MDP profile to be used with the application.</p> <p>This should be the name of a MDP profile defined in the MDP profile file (see Section 7.2.4.3).</p> <p>This MDP profile takes precedence over the MDP profile associated with the node in the scenario configuration file, if any.</p> <p>Note: This parameter can be included only if the parameter MDP-ENABLED is also included.</p> <p>Note: If parameter MDP-ENABLED is included but parameter MDP-PROFILE is not included, and a MDP profile is specified for the node in the scenario configuration file, then the MDP profile specified for the node is used.</p> <p>If parameter MDP-ENABLED is included but parameter MDP-PROFILE is not included, and a MDP profile is not specified for the node in the scenario configuration file, then default MDP values for the MDP profile parameters are used (see Section 7.2.4.3).</p> |
| APPLICATION-NAME <i><application-name></i> <i>Optional</i> | String | Name of the CBR session. This name is printed in the statistics file and statistics database. |

- Notes:**
1. If *<items-to-send>* and *<end-time>* are both greater than 0, packets are transmitted until *<items-to-send>* packets have been sent, *<end-time>* is reached, or the simulation ends, whichever comes first.
 2. At most one of the three parameters PRECEDENCE, DSCP, and TOS can be specified. If PRECEDENCE, DSCP, and TOS are not specified, PRECEDENCE 0 is used as default.

Scenario Configuration File Parameters

[Table 8-2](#) describes the CBR parameters that can be specified in the scenario configuration (.config) file.

TABLE 8-2. CBR Scenario Configuration File Parameters

| Parameter | Value | Description |
|--|--|---|
| APPLICATION-STATISTICS <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Indicates whether statistics collection is enabled for applications (including CBR). |
| TRACE-CBR <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> YES | Indicates whether packet tracing is enabled for CBR. Note: To enable packet tracing, some other parameters need to be configured as well. Refer to Section 4.2.10 of <i>QualNet User's Guide</i> for details. |

Examples of Parameter Usage

The following are examples of CBR configuration:

1. Node 1 sends to node 2 ten items of 1460 bytes each, starting at the beginning of the simulation and up to 600 seconds into the simulation. The inter-packet time is 3 seconds.

```
CBR 1 2 10 1460 3S 0S 600S
```

2. Node 1 continually sends to node 2 items of 1460 bytes each, starting at the beginning of the simulation and ending at 600 seconds into the simulation. The inter-packet time is 1 second. The total number of items sent is 600.

```
CBR 1 2 0 1460 1S 0S 600S
```

3. Node 1 continually sends to node 2 items of 1460 bytes each, starting at the beginning of the simulation. Packets are sent until the end of the simulation. The inter-packet time is 1 second.

```
CBR 1 2 0 1460 1S 0S 0S
```

4. Node 1 continually sends to node 2 items of 1460 bytes each, starting at the beginning of the simulation. Packets are sent until the end of the simulation. The inter-packet time is 1 second. RSVP-TE is used as the transport protocol instead of UDP.

```
CBR 1 2 0 1460 1S 0S 0S RSVP-TE
```

5. Node 1 sends to the node with IP address 192.168.0.8 10 items of 1460 bytes, starting at 5 seconds into the simulation and up to 600 seconds into the simulation. The inter-packet time is 1 second. The Precedence value for each packet is set to 5. The total number of items sent is 595.

```
CBR 1 192.168.0.8 10 1460 1S 5S 600S PRECEDENCE 5
```

6. The node with IP address 192.168.0.1 sends to node 4 10 items of 1460 bytes each, starting at 5 seconds into the simulation and up to 600 seconds into the simulation. The inter-packet time is 1 second. The DSCP value for each packet is set to 40 (express forwarding). The total number of items sent is 595.

```
CBR 192.168.0.1 4 10 1460 1S 5S 600S DSCP 40
```

7. The node with IP address 192.168.0.1 sends to the node with IP address 192.168.0.8 10 items of 1460 bytes each, starting at 5 seconds into the simulation and up to 600 seconds into the simulation. The inter-packet time is 1 second. The TOS value for each packet is set to 175. The total number of items sent is 595.

```
CBR 192.168.0.1 192.168.0.8 10 1460 1S 5S 600S TOS 175
```

8. Node 1 sends to the node with IP address 192.168.0.8 ten items of 1460 bytes each, starting at 5 seconds into the simulation and up to 600 seconds into the simulation. The inter-packet time is 1 second. The TOS value of each item is set to 175. RSVP-TE is used as the transport protocol instead of UDP. The total number of items sent is 595.

```
CBR 1 192.168.0.8 10 1460 1S 5S 600S TOS 175 RSVP-TE
```

9. This is the same as the previous example, except that CBR runs with MDP and uses the MDP profile associated with node 1. If there is no MDP profile associated with node 1, default values for MDP profile parameters are used.

```
CBR 1 192.168.0.8 10 1460 1S 5S 600S TOS 175 RSVP-TE MDP-ENABLED
```

10. This is the same as the previous example, except that MDP uses the user-defined MDP profile `profile-1`.

```
CBR 1 192.168.0.8 10 1460 1S 5S 600S TOS 175 RSVP-TE MDP-ENABLED  
MDP-PROFILE profile-1
```

8.1.4 GUI Configuration

CBR is configured using the CBR Properties Editor.

To configure a CBR session between two nodes, perform the following steps:

1. Click the **CBR** button in the **Applications** tab of the Standard Toolset.
2. On the canvas, click on the source node, drag the mouse to the destination node, and release.
3. Open the CBR Properties Editor by doing one of the following:
 - a. Right-click in the application link on the canvas and select **Properties** from the menu.

- b. In the **Applications** tab of Table View either double-click on the application row or right-click on the application row and select **Properties** from the menu.
4. Set the parameters listed in Table 8-3.

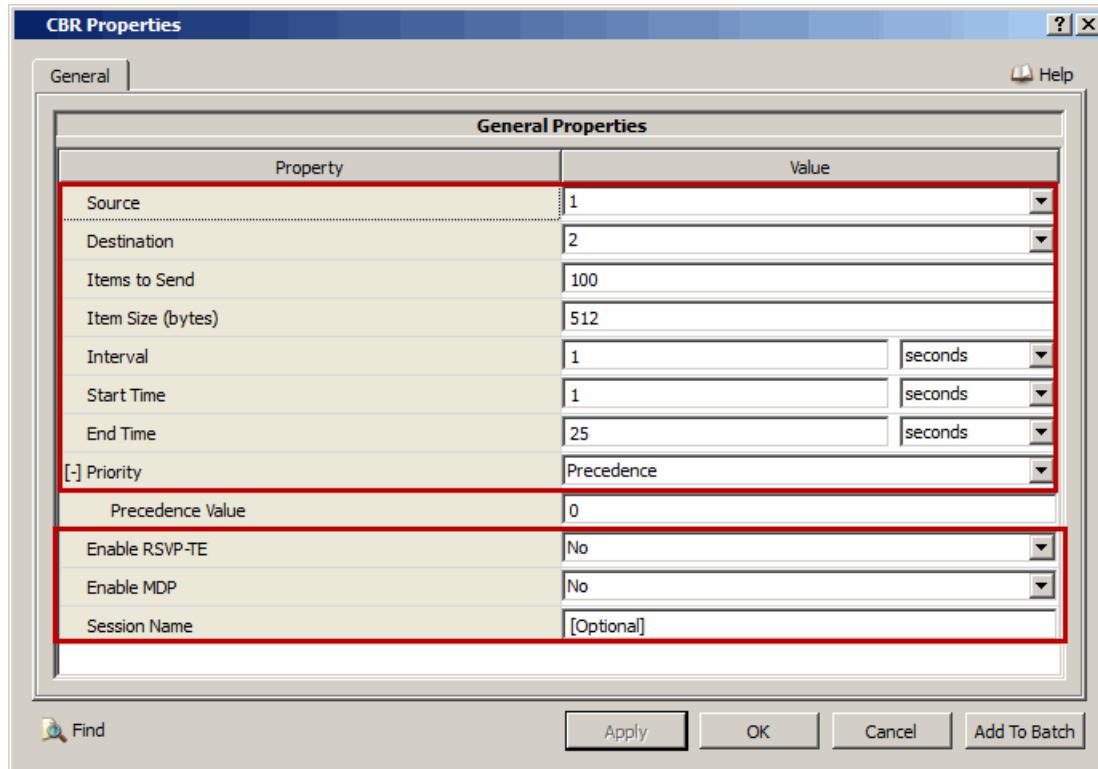


FIGURE 8-1. Setting CBR Parameters

TABLE 8-3. Command Line Equivalent of CBR Parameters

| GUI Parameter | Command Line Parameter |
|------------------------------|-------------------------------------|
| Source | <src> |
| Destination | <dest> |
| Items to Send | <items-to-send> |
| Item Size | <item-size> |
| Start Time | <start-time> |
| End Time | <end-time> |
| Interval | <interval> |
| Priority (set to DSCP) | DSCP |
| Priority (set to Precedence) | PRECEDENCE |
| Priority (set to TOS) | TOS |
| Enable RSVP-TE (set to Yes) | RSVP-TE |
| Enable MDP (set to Yes) | MDP-ENABLED |
| Session Name | APPLICATION-NAME <application-name> |

Setting Parameters

- To specify an IP address as the source (destination) ID, set **Source (Destination)** to one of the IP addresses listed in the drop-down list.
 - To set DSCP, Precedence, or TOS bits for packets, set **Priority** to *DSCP*, *Precedence*, or *TOS*, respectively.
 - To enable MDP, set **Enable MDP** to Yes.
5. If **Priority** is set to *DSCP*, *Precedence*, or *TOS*, set the appropriate dependent parameter listed in Table 8-4. Figure 8-2 shows how to set the dependent parameter when **Priority** is set to *DSCP*. Setting dependent parameters for the other two options is similar.

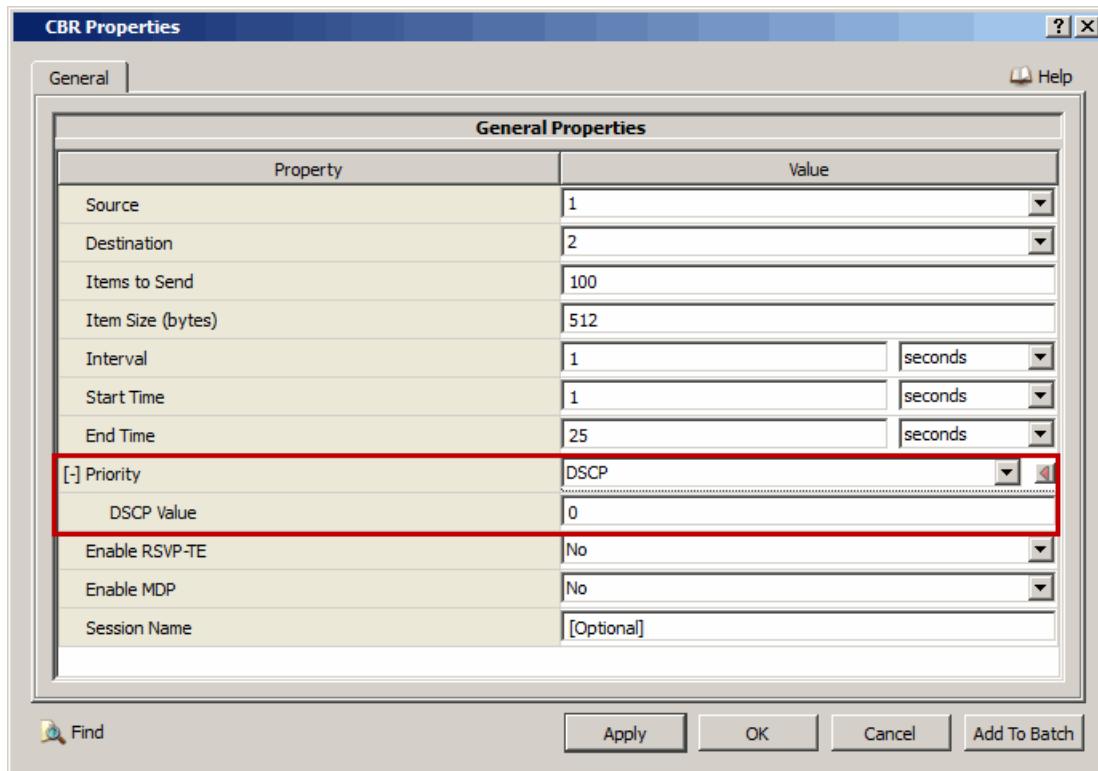


FIGURE 8-2. Setting DSCP Value

TABLE 8-4. Command Line Equivalent of Priority Parameters

| GUI Parameter | Command Line Parameter |
|------------------|------------------------|
| DSCP Value | <dscp-value> |
| Precedence Value | <precedence-value> |
| TOS Value | <tos-value> |

6. If **Enable MDP** is set to Yes, then set the parameters listed in [Table 8-5](#).

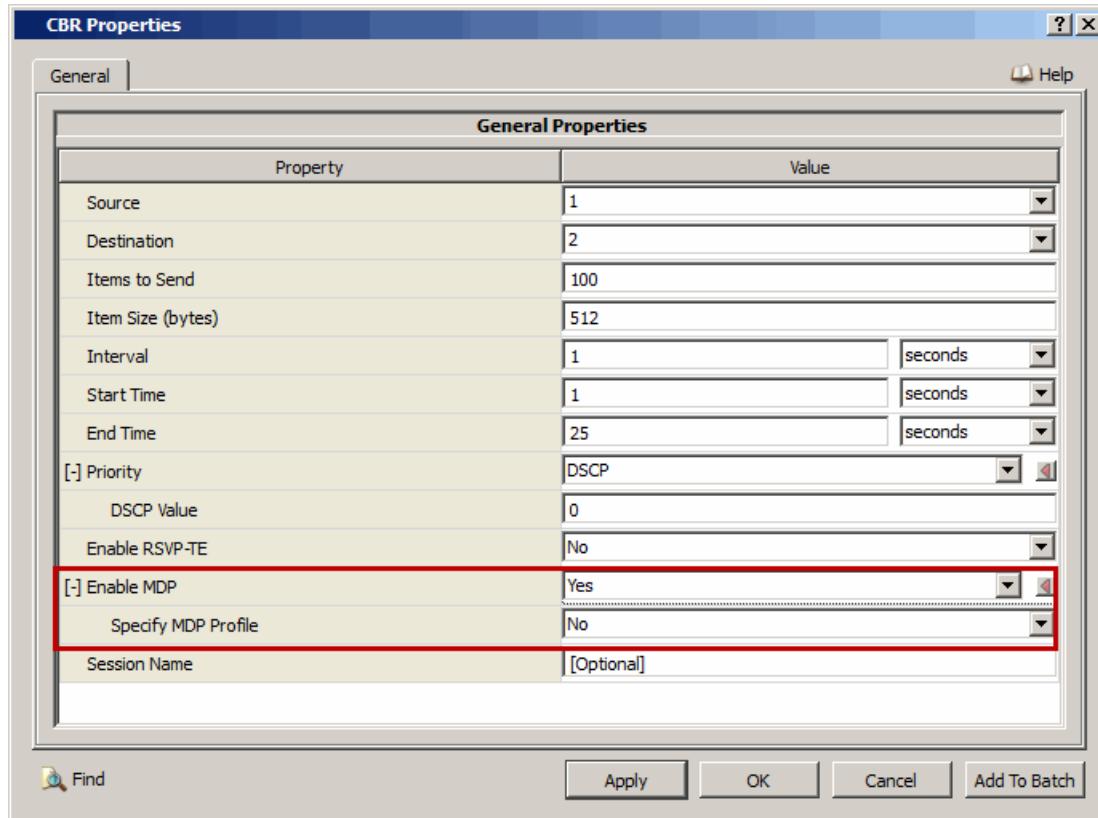


FIGURE 8-3. Enabling MDP

TABLE 8-5. Command Line Equivalent of MDP Parameters

| GUI Parameter | Command Line Parameter |
|----------------------------------|------------------------|
| Specify MDP Profile (set to Yes) | MDP-PROFILE |

Setting Parameters

- To specify an MDP profile, set **Specify MDP Profile** to Yes. To use the default MDP profile, set **Specify MDP Profile** to No.

7. If **Specify MDP Profile** is set to Yes, then set the parameters listed in [Table 8-6](#).

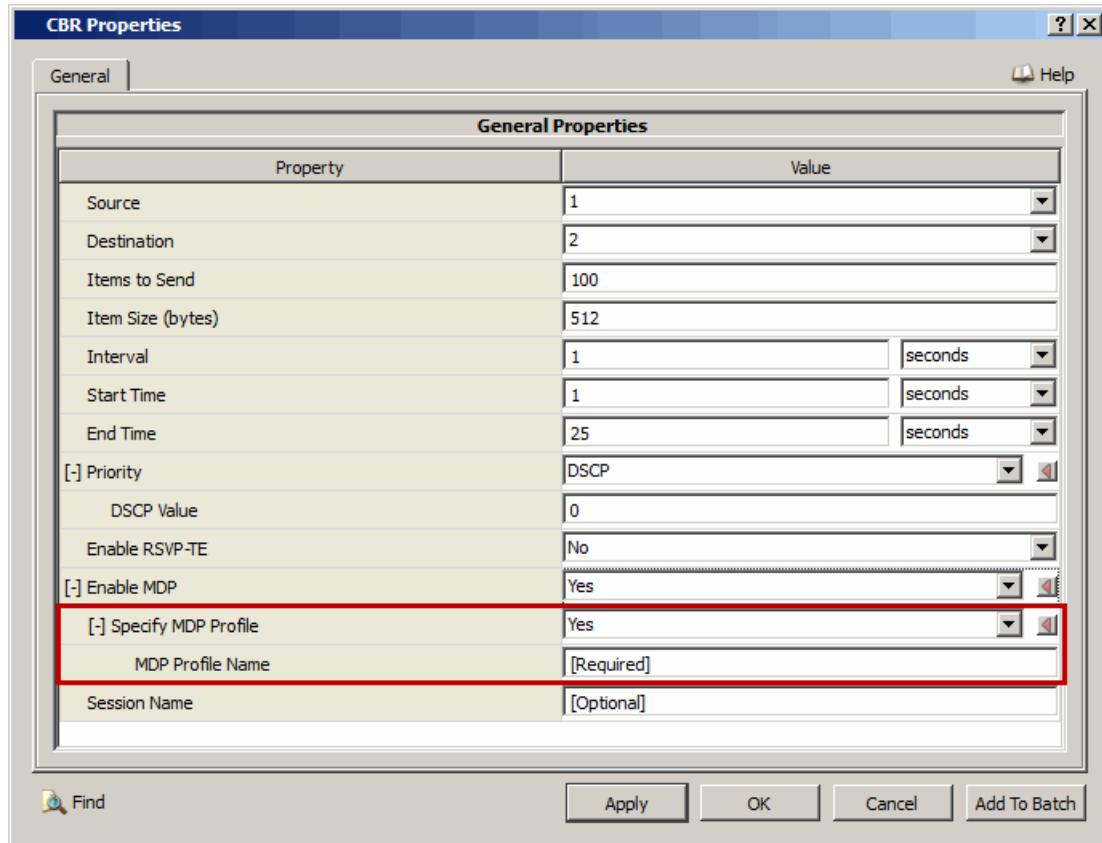


FIGURE 8-4. Specifying MDP Profile

TABLE 8-6. Command Line Equivalent of MDP Profile Parameters

| GUI Parameter | Command Line Parameter |
|------------------|------------------------|
| MDP Profile Name | <profile-name> |

Configuring Statistics Parameters

Statistics for applications (including CBR) can be collected at the global and node levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for CBR, check the box labeled **Application** in the appropriate properties editor.

TABLE 8-7. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| Application | Global, Node | APPLICATION-STATISTICS |

Configuring Packet Tracing Parameters

Packet tracing for CBR can be enabled at the global and node levels. To enable packet tracing for CBR, in addition to setting the CBR trace parameter, **Trace CBR**, several other trace parameters also need to be set. See Section 4.2.10 of *QualNet User's Guide* for details of configuring packet tracing parameters.

TABLE 8-8. Command Line Equivalent of Packet Tracing Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| Trace CBR | Global, Node | TRACE-CBR |

8.1.5 Statistics

This section describes the file, database, and dynamic statistics of the CBR model.

8.1.5.1 File Statistics

[Table 8-9](#) lists the CBR statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 8-9. CBR Statistics

| Statistic | Description |
|--|---|
| CBR Client Statistics | |
| Server address | Server address for CBR Client |
| Session status | Status of the session (open or closed) at the end of simulation |
| Unicast Session Start (seconds) | Time in seconds, when unicast session was started. |
| Unicast Session Finish (seconds) | Time in seconds, when unicast session was finished. |
| First Unicast Fragment Sent (seconds) | Time in seconds, when first unicast fragment was sent. |
| Last Unicast Fragment Sent (seconds) | Time in seconds, when last unicast fragment was sent. |
| Total Unicast Fragments Sent (fragments) | Total number of unicast fragments sent. |
| First Unicast Message Sent (seconds) | Time in seconds, when first unicast message was sent. |
| Last Unicast Message Sent (seconds) | Time in seconds, when last unicast message was sent. |
| Total Unicast Messages Sent (messages) | Total number of unicast messages sent. |
| Total Unicast Data Sent (bytes) | Total number of unicast data bytes sent. |
| Total Unicast Overhead Sent (bytes) | Total number of unicast overhead bytes sent. |
| Unicast Offered Load (bits/second) | Unicast offered load. |
| CBR Server Statistics | |
| Client address | Address of the client |
| Session status | Current status of the session. |
| Unicast Session Finish (seconds) | Unicast Session Finish (seconds) |
| Unicast Session Finish (seconds) | Time in seconds, when unicast session was finished. |
| First Unicast Fragment Received (seconds) | Time in seconds, when first unicast fragment was received. |
| Last Unicast Fragment Received (seconds) | Time in seconds, when last unicast fragment was received. |
| Total Unicast Fragments Received (fragments) | Total number of unicast fragments received. |
| First Unicast Message Received (seconds) | Time in seconds, when first unicast message was received. |
| Last Unicast Message Received (seconds) | Time in seconds, when last unicast message was received. |
| Total Unicast Messages Received (messages) | Total number of unicast messages received. |

TABLE 8-9. CBR Statistics (Continued)

| Statistic | Description |
|--|---|
| Total Unicast Data Received (bytes) | Total number of unicast data bytes received. |
| Total Unicast Overhead Received (bytes) | Total number of unicast overhead bytes received. |
| Average Unicast End-to-End Delay (seconds) | Average delay for packet transmission between client and server (seconds). See note 2. |
| Unicast Received Throughput (bits/second) | Unicast throughput at the server (bits/second). See note 1. |
| Average Unicast Jitter (seconds) | Average jitter for packet transmission between client and server (seconds). See note 3. |

Notes: 1. The throughput at the server is calculated as follows:

- If the session is complete, throughput = (total bytes received * 8) / (time last packet received - time first packet received), where the times are in seconds.
 - If the session is incomplete, throughput = (total bytes received * 8) / (simulation time - time first packet received), where the times are in seconds.
2. Average end-to-end delay = (total of transmission delays of all received packets) / (number of packets received),
where,
transmission delay of a packet = time packet received at server - time packet transmitted at client, where the times are in seconds.
3. Average jitter = (total packet jitter for all received packets) / (number of packets received - 1)
where,
packet jitter = transmission delay of the current packet - transmission delay of the previous packet.
Jitter can be calculated only if at least two packets have been received.

8.1.5.2 Database Statistics

In addition to the file statistics, the CBR model also enters statistics in various scenario statistics database tables. Refer to *QualNet Statistics Database User's Guide* for details.

8.1.5.3 Dynamic Statistics

No dynamic statistics are supported for the CBR model.

8.2 File Transfer Protocol (FTP)

The QualNet FTP application is based on the RFC 959 standard.

8.2.1 Description

FTP represents the File Transfer Protocol server and client. The size of the items sent is taken from network traces, with distributions taken from the TCPlib library. If the number of items to send is specified as 0, then TCPlib also determines the number of items to send.

8.2.2 Command Line Configuration

Application Configuration File Parameters

To specify the FTP application in the application configuration (.app) file, use the following format:

```
FTP      <src> <dest> <items to send> <start time>
        [APPLICATION-NAME <application-name>]
```

Note: All parameters should be entered on the same line.

Table 8-10 shows the FTP parameters and their descriptions. See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 8-10. FTP Parameters

| Parameter | Value | Description |
|---|------------------------------|--|
| <src> <i>Required</i> | Integer or IP Address | Client node's ID or IP address. |
| <dest> <i>Required</i> | Integer or IP Address | Destination node's ID or IP address. |
| <items-to-send> <i>Required</i> | Integer <i>Range: ≥ 0</i> | Number of packets to send. If this is specified as 0, the number of items to send is determined by TCPlib. See note 2. |
| <start-time> <i>Required</i> | Time <i>Range: ≥ 0S</i> | Time when the transmission of packets should begin. |
| APPLICATION-NAME <application-name> <i>Optional</i> | String | Name of the FTP session. This name is printed in the statistics file and statistics database. |

- Notes:**
1. If you want to be able to configure the size of packets, you should use FTP/Generic, rather than FTP. FTP/Generic represents a more configurable model of the File Transfer Protocol. The size of the items sent is not determined by network traces, instead it is user-specified.
 2. If <items-to-send> is greater than 0, packets are transmitted until <items-to-send> packets have been sent, or when the simulation ends, whichever comes first.

Scenario Configuration File Parameters

Table 8-11 describes the FTP parameters that can be specified in the scenario configuration (.config) file.

TABLE 8-11. FTP Scenario Configuration File Parameters

| Parameter | Value | Description |
|--|--|---|
| APPLICATION-STATISTICS <i>Optional</i> Scope: Global, Node | List: • YES • NO <i>Default: NO</i> | Indicates whether statistics are collected for application protocols (including FTP). |

Examples of Parameter Usage

The following are examples of FTP configuration:

```
FTP 1 2 10 OS
```

Node 1 sends node 2 ten items at the start of the simulation, with the size of each item randomly determined by tcplib.

```
FTP 1 2 0 100S
```

Node 1 sends node 2 the number of items randomly picked by tcplib after 100 seconds into the simulation. The size of each item is also randomly determined by tcplib.

8.2.3 GUI Configuration

FTP is configured using the FTP Properties Editor.

To configure a FTP session between two nodes, perform the following steps:

1. Click the **FTP** button in the **Applications** tab of the Standard Toolset.
2. On the canvas, click on the source node, drag the mouse to the destination node, and release.
3. Open the FTP Properties Editor by doing one of the following:
 - a. Right-click in the application link on the canvas and select **Properties** from the menu.
 - b. In the **Applications** tab of Table View, either double-click on the application row or right-click on the application row and select **Properties** from the menu.

4. Set the parameters listed in Table 8-12.

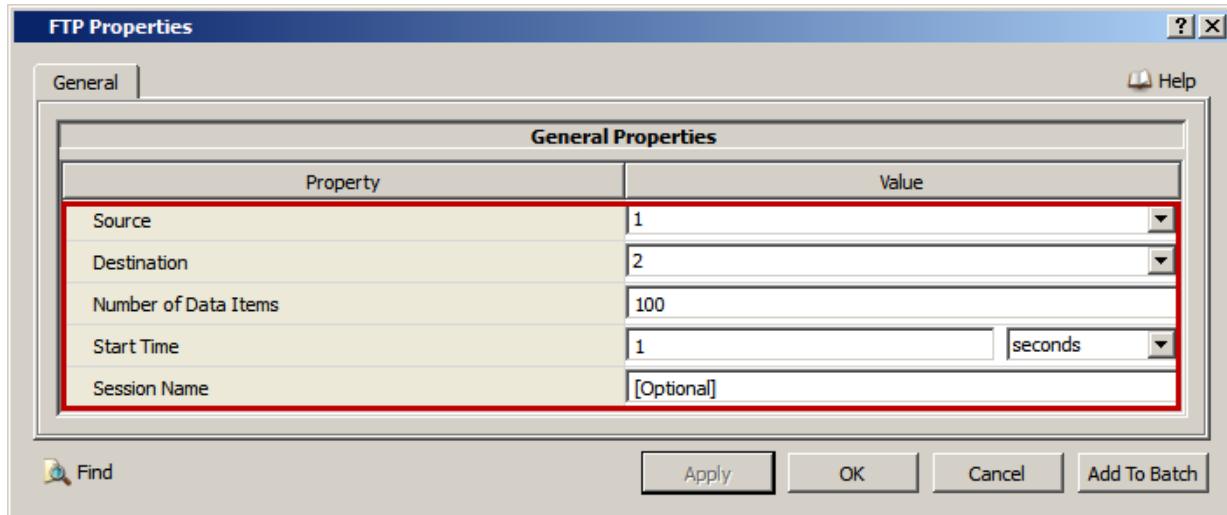


FIGURE 8-5. Setting FTP Parameters

TABLE 8-12. Command Line Equivalent of FTP Parameters

| GUI Parameter | Command Line Parameter |
|----------------------|-------------------------------------|
| Source | <src> |
| Destination | <dest> |
| Number of Data Items | <items-to-send> |
| Start Time | <start-time> |
| Session Name | APPLICATION-NAME <application-name> |

Setting Parameters

- To specify an IP address as the source (destination) ID, set **Source (Destination)** to one of the IP addresses listed in the drop-down list.

Configuring Statistics Parameters

Statistics for applications (including FTP) can be collected at the global and node levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for FTP, check the box labeled **Application** in the appropriate properties editor.

TABLE 8-13. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| Application | Global, Node | APPLICATION-STATISTICS |

8.2.4 Statistics

This section describes the file, database, and dynamic statistics of the FTP model.

8.2.4.1 File Statistics

[Table 8-14](#) lists the FTP statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 8-14. FTP Statistics

| Statistic | Description |
|--|--|
| FTP Client Statistics | |
| Server Address | Specifies the server address |
| Session Status | Specifies whether the session status is open or closed. |
| Unicast Session Start (seconds) | Time in seconds, when unicast session was started. |
| Unicast Session Finish (seconds) | Time in seconds, when unicast session was finished. |
| First Unicast Fragment Sent (seconds) | Time in seconds, when first unicast fragment was sent. |
| Last Unicast Fragment Sent (seconds) | Time in seconds, when last unicast fragment was sent. |
| First Unicast Fragment Received (seconds) | Time in seconds, when first unicast fragment was received. |
| Last Unicast Fragment Received (seconds) | Time in seconds, when last unicast fragment was received. |
| Total Unicast Fragments Sent (fragments) | Total number of unicast fragments sent. |
| Total Unicast Fragments Received (fragments) | Total number of unicast fragments received. |
| First Unicast Message Sent (seconds) | Time in seconds, when first unicast message was sent. |
| Last Unicast Message Sent (seconds) | Time in seconds, when last unicast message was sent. |
| First Unicast Message Received (seconds) | Time in seconds, when first unicast message was received. |
| Last Unicast Message Received (seconds) | Time in seconds, when last unicast message was received. |
| Total Unicast Messages Sent (messages) | Total number of unicast messages sent. |
| Total Unicast Messages Received (messages) | Total number of unicast messages received. |
| Total Unicast Data Sent (bytes) | Total number of data bytes sent. |
| Total Unicast Data Received (bytes) | Total number of data bytes received. |
| Total Unicast Overhead Sent (bytes) | Total number of overhead bytes sent. |
| Total Unicast Overhead Received (bytes) | Total number of overhead bytes received. |
| Average Unicast End-to-End Delay (seconds) | Average unicast end-to-end delay. |
| Unicast Offered Load (bits/second) | Unicast offered load. |
| Unicast Received Throughput (bits/second) | Specifies the total throughput of the server (bits/seconds). See Note 2. |
| Smoothed Unicast Jitter (seconds) | Smoothed unicast jitter. |
| Average Unicast Jitter (seconds) | Average unicast jitter. |
| Total Unicast Jitter (seconds) | Total unicast jitter. |
| FTP Server Statistics | |
| Client Address | Specifies the client address |
| Unicast Session Start (seconds) | Time in seconds, when unicast session was started. |
| Unicast Session Finish (seconds) | Time in seconds, when unicast session was finished. |
| First Unicast Fragment Sent (seconds) | Time in seconds, when first unicast fragment was sent. |
| Last Unicast Fragment Sent (seconds) | Time in seconds, when last unicast fragment was sent. |
| First Unicast Fragment Received (seconds) | Time in seconds, when first unicast fragment was received. |

TABLE 8-14. FTP Statistics

| Statistic | Description |
|--|---|
| Last Unicast Fragment Received (seconds) | Time in seconds, when last unicast fragment was received. |
| Total Unicast Fragments Sent (fragments) | Total number of unicast fragments sent. |
| Total Unicast Fragments Received (fragments) | Total number of unicast fragments received. |
| First Unicast Message Sent (seconds) | Time in seconds, when first unicast message was sent. |
| Last Unicast Message Sent (seconds) | Time in seconds, when last unicast message was sent. |
| First Unicast Message Received (seconds) | Time in seconds, when first unicast message was received. |
| Last Unicast Message Received (seconds) | Time in seconds, when last unicast message was received. |
| Total Unicast Messages Sent (messages) | Total number of unicast messages sent. |
| Total Unicast Messages Received (messages) | Total number of unicast messages received. |
| Total Unicast Data Sent (bytes) | Total number of data bytes sent. |
| Total Unicast Data Received (bytes) | Total number of data bytes received. |
| Total Unicast Overhead Sent (bytes) | Total number of overhead bytes sent. |
| Total Unicast Overhead Received (bytes) | Total number of overhead bytes received. |
| Average Unicast End-to-End Delay (seconds) | Average unicast end-to-end delay. |
| Unicast Offered Load (bits/second) | Unicast offered load. |
| Unicast Received Throughput (bits/second) | Specifies the total throughput of the server (bits/second). See Note 1. |
| Smoothed Unicast Jitter (seconds) | Smoothed unicast jitter. |
| Average Unicast Jitter (seconds) | Average unicast jitter. |
| Total Unicast Jitter (seconds) | Total unicast jitter. |

Notes: 1. The throughput at the server is calculated as follows:

- If the session is complete, throughput = (total bytes sent * 8) / (time last packet received - time first packet received), where the times are in seconds.
- If the session is incomplete, throughput = (total bytes sent * 8) / (simulation time - time first packet received), where the times are in seconds.

2. The throughput at the client is calculated as follows:

- If the session is complete, i.e., if all packets have been sent before the simulation ends, throughput = (total bytes received * 8) / (time last packet sent - time first packet sent), where the times are in seconds.
- If the session is incomplete, i.e., if all packets have not been sent before the simulation ends, throughput = (total bytes received * 8) / (simulation time - time first packet sent), where the times are in seconds.

8.2.4.2 Database Statistics

In addition to the file statistics, the FTP model also enters statistics in various scenario statistics database tables. Refer to *QualNet Statistics Database User's Guide* for details.

8.2.4.3 Dynamic Statistics

No dynamic statistics are supported for the FTP model.

8.2.5 References

1. RFC 959, "File Transfer Protocol (FTP)" J. Postel, J. Reynolds. October 1985.

8.3 File Transfer Protocol/Generic (FTP/Generic)

8.3.1 Description

FTP/Generic represents a more configurable model of the File Transfer Protocol. The size of the items sent is not determined by network traces, instead it is user-specified.

8.3.2 Command Line Configuration

Application Configuration File Parameters

To specify FTP/Generic traffic, include the following statement in the application configuration (.app) file:

```
FTP/GENERIC <src> <dest> <items-to-send> <item-size>
    <start-time> <end-time>
    [TOS <tos-value> | DSCP <dscp-value> |
     PRECEDENCE <precedence-value>]
    [APPLICATION-NAME <application-name>]
```

Note: All parameters should be entered on the same line.

Table 8-15 shows the parameters for the FTP/Generic model. See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 8-15. FTP/Generic Parameters

| Parameter | Value | Description |
|------------------------------------|---|--|
| <src> <i>Required</i> | Integer or IP Address | Client node's ID or IP address. |
| <dest> <i>Required</i> | Integer or IP Address | Server node's ID or IP address. |
| <items-to-send> <i>Required</i> | Integer <i>Range:</i> ≥ 0 <i>Unit:</i> bytes | Number of packets to send. If this is set to 0, items will be sent continually until <end-time> or until the end of the simulation, whichever comes first. See note 1. |
| <item-size> <i>Required</i> | Integer <i>Range:</i> > 0 <i>Unit:</i> bytes | Size of each item. |
| <start-time> <i>Required</i> | Time <i>Range:</i> $\geq 0\text{S}$ | Time when the transmission of packets should begin. |

TABLE 8-15. FTP/Generic Parameters (Continued)

| Parameter | Value | Description |
|---|--|---|
| <end-time> <i>Required</i> | Time <i>Range:</i> $\geq 0\text{S}$ | Time when the transmission of packets should end. If this is set to 0, transmission ends after <items-to-send> packets have been sent or until the end of simulation, whichever comes first. Note: <end-time> should be greater than <start-time> or equal to 0. |
| TOS <tos-value> <i>Optional</i> | Integer <i>Range:</i> [0, 255] | Value of the 8-bit TOS field of the IP header for the packets generated. See note 2. |
| DSCP <dscp-value> <i>Optional</i> | Integer <i>Range:</i> [0, 63] | Value of the 6-bit DSCP field of the IP header for the packets generated. See note 2. |
| PRECEDENCE <precedence-value> <i>Optional</i> | Integer <i>Range:</i> [0, 7] | Value of the 3-bit Precedence field of the IP header for the packets generated. See note 2. |
| APPLICATION-NAME <application-name> <i>Optional</i> | String | Name of the FTP/Generic session. This name is printed in the statistics file and statistics database. |

- Notes:**
1. If <items-to-send> and <end-time> are both greater than 0, FTP/Generic will run until either <items-to-send> is done, <end-time> is reached, or the simulation ends, whichever comes first.
 2. At most one of the three parameters PRECEDENCE, DSCP, and TOS can be specified. If PRECEDENCE, DSCP or TOS is not specified, PRECEDENCE 0 is used as default.

Scenario Configuration File Parameters

[Table 8-16](#) describes the FTP/Generic parameters that can be specified in the scenario configuration (.config) file.

TABLE 8-16. FTP/Generic Scenario Configuration File Parameters

| Parameter | Value | Description |
|--|--|---|
| APPLICATION-STATISTICS <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Indicates whether statistics are collected for application protocols (including FTP/Generic). |
| TRACE-GEN-FTP <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> YES | Indicates whether packet tracing is enabled for FTP Generic. Note: To enable packet tracing, some other parameters need to be configured as well. Refer to Section 4.2.10 of <i>QualNet User's Guide</i> for details. |

Examples of Parameter Usage

The following are examples of FTP/Generic configuration:

1. Node 1 sends node 2 up to 10 items of 1460 bytes each, starting at the beginning of the simulation and up to 600 seconds into the simulation. At most 10 packets are sent and no packets are sent after 600 seconds have elapsed, even if fewer than 10 packets have been sent.

```
FTP/GENERIC 1 2 10 1460 0S 600S
```

2. Node 1 continually sends to node 2 items of 1460 bytes each, starting at the beginning of the simulation until the end of simulation.

```
FTP/GENERIC 1 2 0 1460 0S 0S
```

3. Node 1 sends ten 1460-byte items to node 2 with IP Precedence of 4 for each data item, starting at the beginning of the simulation and up to 600 seconds into the simulation. At most 10 packets are sent and no packets are sent after 600 seconds have elapsed, even if fewer than 10 packets have been sent.

```
FTP/GENERIC 1 2 10 1460 0S 600S PRECEDENCE 4
```

4. Node 1 sends fifteen 1460-byte items to node 2 with IP DSCP of 40 for each data item, starting at the beginning of the simulation and up to 1000 seconds into the simulation. At most 15 packets are sent and no packets are sent after 1000 seconds have elapsed, even if fewer than 15 packets have been sent.

```
FTP/GENERIC 1 2 15 1460 0S 1000S DSCP 40
```

5. Node 1 sends fifty 1460-byte items to node 2 with IP TOS of 170 for each data item, starting at the beginning of the simulation and up to 500 seconds into the simulation. At most 50 packets are sent and no packets are sent after 500 seconds have elapsed, even if fewer than 50 packets have been sent.

```
FTP/GENERIC 1 2 50 1460 0S 500S TOS 170
```

8.3.3 GUI Configuration

FTP Generic is configured using the FTP Generic Properties Editor.

To configure a FTP Generic session between two nodes, perform the following steps:

1. Click the **FTP GEN** button in the **Applications** tab of the Standard Toolset.
2. On the canvas, click on the source node, drag the mouse to the destination node, and release.
3. Open the FTP Generic Properties Editor by doing one of the following:
 - c. Right-click in the application link on the canvas and select **Properties** from the menu.
 - d. In the **Applications** tab of Table View, either double-click on the application row or right-click on the application row and select **Properties** from the menu.
4. Set the parameters listed in [Table 8-17](#).

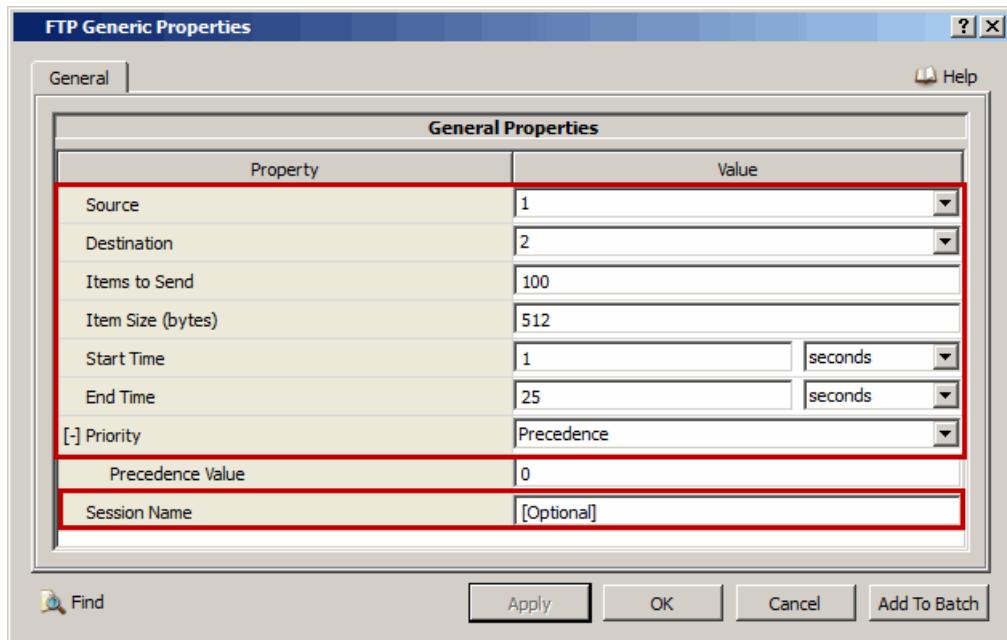


FIGURE 8-6. Setting FTP Generic Parameters

TABLE 8-17. Command Line Equivalent of FTP Generic Parameters

| GUI Parameter | Command Line Parameter |
|---------------|------------------------|
| Source | <src> |
| Destination | <dest> |

TABLE 8-17. Command Line Equivalent of FTP Generic Parameters (Continued)

| GUI Parameter | Command Line Parameter |
|--------------------------------------|-------------------------------------|
| Items to Send | <items-to-send> |
| Item Size | <item-size> |
| Start Time | <start-time> |
| End Time | <end-time> |
| Priority (set to <i>DSCP</i>) | DSCP |
| Priority (set to <i>Precedence</i>) | PRECEDENCE |
| Priority (set to <i>TOS</i>) | TOS |
| Session Name | APPLICATION-NAME <application-name> |

Setting Parameters

- To specify an IP address as the source (destination) ID, set **Source (Destination)** to one of the IP addresses listed in the drop-down list.
 - To set DSCP, Precedence, or TOS bits for packets, set **Priority** to *DSCP*, *Precedence*, or *TOS*, respectively.
5. If **Priority** is set to *DSCP*, *Precedence*, or *TOS*, set the appropriate dependent parameter listed in Table 8-18. Figure 8-7 shows how to set the dependent parameter when **Priority** is set to *DSCP*. Setting dependent parameters for the other two options is similar.

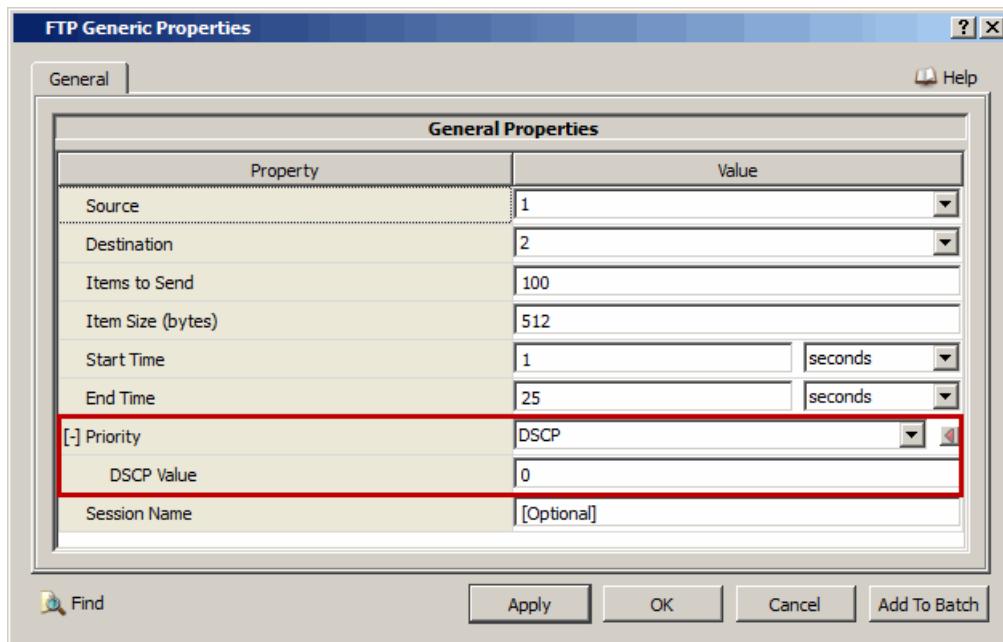
**FIGURE 8-7. Setting DSCP Bits**

TABLE 8-18. GUI Parameters for Specifying QoS for FTP Generic

| GUI Parameter | Command Line Parameter |
|------------------|------------------------|
| DSCP Value | <dscp-value> |
| Precedence Value | <precedence-value> |
| TOS Value | <tos-value> |

Configuring Statistics Parameters

Statistics for applications (including FTP Generic) can be collected at the global and node levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for FTP Generic, check the box labeled **Application** in the appropriate properties editor.

TABLE 8-19. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| Application | Global, Node | APPLICATION-STATISTICS |

8.3.4 Statistics

This section describes the file, database, and dynamic statistics of the FTP/Generic model.

8.3.4.1 File Statistics

[Table 8-20](#) list the FTP/Generic statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 8-20. FTP/Generic Statistics

| Statistic | Description |
|--|--|
| FTP/Generic Client | |
| Server Address | Specifies the server address |
| Session Status | Status of the session (open or closed) at the end of simulation. |
| Unicast Session Start (seconds) | Time in seconds, when unicast session was started. |
| Unicast Session Finish (seconds) | Time in seconds, when unicast session was finished. |
| First Unicast Fragment Sent (seconds) | Time in seconds, when first unicast fragment was sent. |
| Last Unicast Fragment Sent (seconds) | Time in seconds, when last unicast fragment was sent. |
| Total Unicast Fragments Sent (fragments) | Total number of unicast fragments sent. |
| First Unicast Message Sent (seconds) | Time in seconds, when first unicast message was sent. |
| Last Unicast Message Sent (seconds) | Time in seconds, when last unicast message was sent. |
| Total Unicast Messages Sent (messages) | Total number of unicast messages sent. |
| Total Unicast Data Sent (bytes) | Total number of unicast data bytes sent. |
| Total Unicast Overhead Sent (bytes) | Total number of unicast overhead bytes sent. |
| Unicast Offered Load (bits/second) | Unicast offered load. |

TABLE 8-20. FTP/Generic Statistics (Continued)

| Statistic | Description |
|--|--|
| FTP/Generic Server | |
| Client Address | Specifies the client address. |
| Session Status | Status of the session (open or closed) at the end of simulation. |
| Unicast Session Start (seconds) | Time in seconds, when unicast session was started. |
| Unicast Session Finish (seconds) | Time in seconds, when unicast session was finished. |
| First Unicast Fragment Received (seconds) | Time in seconds, when first unicast fragment was received. |
| Last Unicast Fragment Received (seconds) | Time in seconds, when last unicast fragment was received. |
| Total Unicast Fragments Received (fragments) | Total number of unicast fragments received. |
| First Unicast Message Received (seconds) | Time in seconds, when first unicast message was received. |
| Last Unicast Message Received (seconds) | Time in seconds, when last unicast message was received. |
| Total Unicast Messages Received (messages) | Total number of unicast messages received. |
| Total Unicast Data Received (bytes) | Total number of unicast data bytes received. |
| Total Unicast Overhead Received (bytes) | Total number of unicast overhead bytes received. |
| Average Unicast End-to-End Delay (seconds) | Average unicast end-to-end delay. |
| Unicast Received Throughput (bits/second) | Unicast throughput at the server. |
| Average Unicast Jitter (seconds) | Average unicast jitter. |

Note: The throughput at the server is calculated as follows:

- If the session is complete, throughput = (total bytes received * 8) / (time last packet received - time first packet received), where the times are in seconds.
- If the session is incomplete, throughput = (total bytes received * 8) / (simulation time - time first packet received), where the times are in seconds.

8.3.4.2 Database Statistics

In addition to the file statistics, the FTP/Generic model also enters statistics in various scenario statistics database tables. Refer to *QualNet Statistics Database User's Guide* for details.

8.3.4.3 Dynamic Statistics

No dynamic statistics are supported for the FTP/Generic model.

8.4 HyperText Transfer Protocol (HTTP)

The QualNet HTTP model is based on the following standards:

- RFC1945 Hypertext Transfer Protocol -- HTTP/1.0.
- RFC2068 Hypertext Transfer Protocol -- HTTP/1.1.
- RFC2616 Hypertext Transfer Protocol -- HTTP/1.1.

8.4.1 Description

HTTP simulates single-TCP connection web servers and clients. The size of HTTP items retrieved, number of items per Web page, think time, and user browsing behavior are taken from packet traces of HTTP network conversations.

8.4.2 Features and Assumptions

This section describes the implemented features, omitted features, assumptions and limitations of the HTTP model.

8.4.2.1 Implemented Features

- HTTP server initialization, message processing, and finalization functions as described in [4].
- HTTP request and response.

8.4.2.2 Omitted Features

None.

8.4.2.3 Assumptions and Limitations

- The HTTP service response time is determined as described in [4].

8.4.3 Command Line Configuration

Application Configuration File Parameters

Configuring HTTP traffic in the application configuration file (.app) consists of two parts: client configuration and server configuration.

To configure an HTTP client, use the following format:

```
HTTP  <client-ID> <num-servers> <server-1> ... <server-n>
      <start-time> <thresh>
      [APPLICATION-NAME <application-name>]
```

Note: All client parameters should be entered on the same line.

To configure an HTTP server, use the following format:

```
HTTPD  <server-ID>
```

[Table 8-21](#) shows the HTTP parameters for clients and servers. See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 8-21. HTTP Parameters

| Parameter | Value | Description |
|---|------------------------------|---|
| Client Parameters | | |
| <client-ID> <i>Required</i> | Integer <i>Range:</i> > 0 | Node ID of the client. |
| <num-servers> <i>Required</i> | Integer <i>Range:</i> > 0 | Number of servers. |
| <server-i> <i>Required</i> | Integer <i>Range:</i> > 0 | Node ID of the <i>i</i> th server. Note: There should be <num-servers> occurrences of this parameter. A node ID can appear in the list of servers for a client only if it is configured as an HTTP server. |
| <start-time> <i>Required</i> | Time <i>Range:</i> ≥ 0S | Start time from when the client begins requesting pages. |
| <thresh> <i>Required</i> | Time <i>Range:</i> ≥ 0S | Ceiling (specified in units of time) on the amount of “think time” allowed for a client. The network-traced based amount of time modulo this threshold is used to determine think time. |
| Server Parameters | | |
| <server-ID> <i>Required</i> | Integer <i>Range:</i> > 0 | Node ID of the server. |
| APPLICATION-NAME <application-name> <i>Optional</i> | String | Name of the HTTP session. This name is printed in the statistics file and statistics database. |

Scenario Configuration File Parameters

[Table 8-22](#) describes the HTTP parameters that can be specified in the scenario configuration (.config) file.

TABLE 8-22. HTTP Scenario Configuration File Parameters

| Parameter | Value | Description |
|--|---|--|
| APPLICATION-STATISTICS <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Indicates whether statistics are collected for application protocols (including HTTP). |

Examples of Parameter Usage

The following is an example of HTTP configuration:

There are HTTP servers on nodes 2, 5, 8, and 11. There is an HTTP client on node 1. This client chooses between servers 2, 5, and 11 when requesting web pages. It begins browsing after 10 seconds of simulation time have passed, and will remain idle for at most 2 minutes at a time.

```
HTTPD 2
HTTPD 5
HTTPD 8
HTTPD 11
HTTP 1 3 2 5 11 10S 120S
```

8.4.4 GUI Configuration

HTTP is configured using the HTTP Properties Editor.

To configure a HTTP session between two nodes, perform the following steps:

1. Click the **HTTP** button in the **Single Host Applications** tab of the Standard Toolset.
2. On the canvas, click on the node and release.
3. Open the HTTP Properties Editor by doing one of the following:
 - a. On the canvas, either double-click on the application link or right-click on the application link and select **Properties** from the menu.
 - b. In the **Applications** tab of Table View, either double-click on the application row or right-click on the application row and select **Properties** from the menu.
4. Set the parameters listed in [Table 8-23](#).

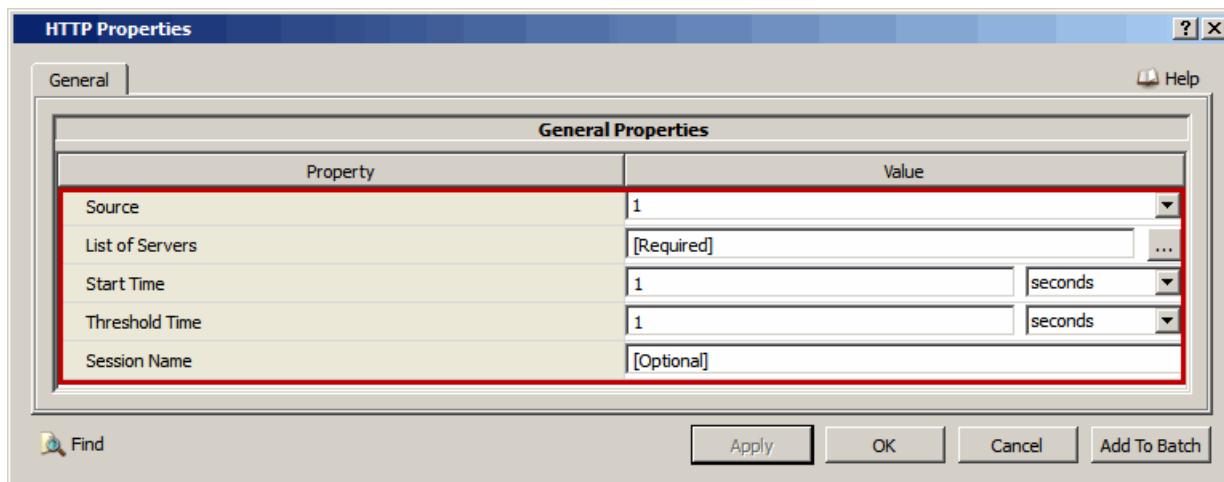


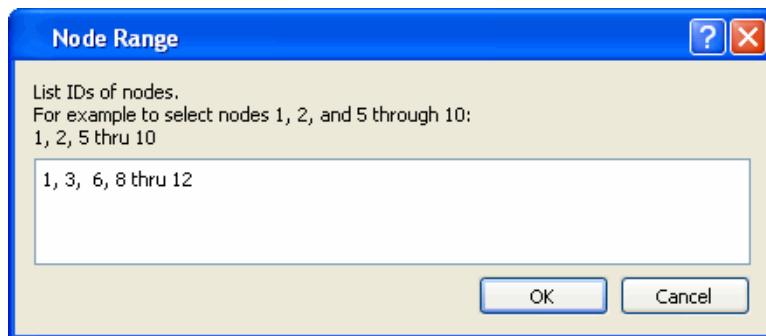
FIGURE 8-8. HTTP Parameters

TABLE 8-23. Command Line Equivalent of HTTP Parameters

| GUI Parameter | Command Line Parameter |
|-----------------|-------------------------------------|
| Source | <client-ID> |
| List of Servers | <server-1> ... <server-n> |
| Start Time | <start-time> |
| Threshold Time | <thresh> |
| Session Name | APPLICATION-NAME <application-name> |

Setting Parameters

- To specify an IP address as the client ID, set **Source** to one of the IP addresses listed in the drop-down list.
5. To select the list of servers, do the following:
- a. Click the **Select Nodes**  button in the **Value** column of **List of Servers**. This opens the dialog to enter node IDs ([Figure 8-9](#)).
 - b. Enter the node IDs of the server nodes.

**FIGURE 8-9. Selecting Server Nodes****Configuring Statistics Parameters**

Statistics for applications (including HTTP) can be collected at the global and node levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters. To enable statistics collection for HTTP, check the box labeled **Application** in the appropriate properties editor.

TABLE 8-24. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| Application | Global, Node | APPLICATION-STATISTICS |

8.4.5 Statistics

This section describes the file, database, and dynamic statistics of the HTTP model.

8.4.5.1 File Statistics

[Table 8-25](#) lists the HTTP statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 8-25. HTTP Statistics

| Statistic | Description |
|---|--|
| HTTP Client | |
| Number of Connections | Total number of connections taken by the client. |
| Average Connection Length (s) | The average time of connection length. (seconds). |
| Average Number of Pages per Connection | Specifies the average number of pages per connection. |
| Average Number of Bytes Received per Connection | Specifies the average number of bytes received per connection. |
| Average Number of Bytes Sent per Connection | Specifies the average number of bytes sent per connection. |
| Average Page Request Time (s) | Specifies the average page request time (seconds). |
| Longest Page Request Time (s) | Specifies the longest page request time (seconds). |
| Unicast Session Start (seconds) | Time in seconds, when unicast session was started. |
| Unicast Session Finish (seconds) | Time in seconds, when unicast session was finished. |
| First Unicast Fragment Sent (seconds) | Time in seconds, when first unicast fragment was sent. |
| Last Unicast Fragment Sent (seconds) | Time in seconds, when last unicast fragment was sent. |
| First Unicast Fragment Received (seconds) | Time in seconds, when first unicast fragment was received. |
| Last Unicast Fragment Received (seconds) | Time in seconds, when last unicast fragment was received. |
| Total Unicast Fragments Sent (fragments) | Total number of unicast fragments sent. |
| Total Unicast Fragments Received (fragments) | Total number of unicast fragments received. |
| First Unicast Message Sent (seconds) | Time in seconds, when first unicast message was sent. |
| Last Unicast Message Sent (seconds) | Time in seconds, when last unicast message was sent. |
| First Unicast Message Received (seconds) | Time in seconds, when first unicast message was received. |
| Last Unicast Message Received (seconds) | Time in seconds, when last unicast message was received. |
| Total Unicast Messages Sent (messages) | Total number of unicast messages sent. |
| Total Unicast Messages Received (messages) | Total number of unicast messages received. |
| Total Unicast Data Sent (bytes) | Total number of unicast data bytes sent. |
| Total Unicast Data Received (bytes) | Total number of unicast data bytes received. |
| Total Unicast Overhead Sent (bytes) | Total number of unicast overhead bytes sent. |
| Total Unicast Overhead Received (bytes) | Total number of unicast overhead bytes received. |
| Average Unicast End-to-End Delay (seconds) | Average unicast end-to-end delay. |
| Unicast Offered Load (bits/second) | Unicast offered load. |
| Unicast Received Throughput (bits/second) | Unicast received throughput. |
| Smoothed Unicast Jitter (seconds) | Smoothed unicast jitter. |
| Average Unicast Jitter (seconds) | Average unicast jitter. |
| Total Unicast Jitter (seconds) | Total unicast jitter. |
| HTTP Server | |
| Client Address | Specifies the client address |

TABLE 8-25. HTTP Statistics (Continued)

| Statistic | Description |
|--|--|
| Connection Length (s) | Specifies the length of the connection (seconds) |
| Pages Sent | Specifies the total number of pages sent |
| Unicast Session Start (seconds) | Time in seconds, when unicast session was started. |
| Unicast Session Finish (seconds) | Time in seconds, when unicast session was finished. |
| First Unicast Fragment Sent (seconds) | Time in seconds, when first unicast fragment was sent. |
| Last Unicast Fragment Sent (seconds) | Time in seconds, when last unicast fragment was sent. |
| First Unicast Fragment Received (seconds) | Time in seconds, when first unicast fragment was received. |
| Last Unicast Fragment Received (seconds) | Time in seconds, when last unicast fragment was received. |
| Total Unicast Fragments Sent (fragments) | Total number of unicast fragments sent. |
| Total Unicast Fragments Received (fragments) | Total number of unicast fragments received. |
| First Unicast Message Sent (seconds) | Time in seconds, when first unicast message was sent. |
| Last Unicast Message Sent (seconds) | Time in seconds, when last unicast message was sent. |
| First Unicast Message Received (seconds) | Time in seconds, when first unicast message was received. |
| Last Unicast Message Received (seconds) | Time in seconds, when last unicast message was received. |
| Total Unicast Messages Sent (messages) | Total number of unicast messages sent. |
| Total Unicast Messages Received (messages) | Total number of unicast messages received. |
| Total Unicast Data Sent (bytes) | Total number of unicast data bytes sent. |
| Total Unicast Data Received (bytes) | Total number of unicast data bytes received. |
| Total Unicast Overhead Sent (bytes) | Total number of unicast overhead bytes sent. |
| Total Unicast Overhead Received (bytes) | Total number of unicast overhead bytes received. |
| Average Unicast End-to-End Delay (seconds) | Average unicast end-to-end delay. |
| Unicast Offered Load (bits/second) | Unicast offered load. |
| Unicast Received Throughput (bits/second) | Unicast received throughput. |
| Smoothed Unicast Jitter (seconds) | Smoothed unicast jitter. |
| Average Unicast Jitter (seconds) | Average unicast jitter. |
| Total Unicast Jitter (seconds) | Total unicast jitter. |

Note: The throughput at the server is calculated as follows:

- If the session is complete, throughput = ((total bytes received + total bytes sent) * 8) / (time last packet received - time first packet received), where the times are in seconds.
- If the session is incomplete, throughput = ((total bytes received + total bytes sent) * 8) / (simulation time - time first packet received), where the times are in seconds.

8.4.5.2 Database Statistics

In addition to the file statistics, the HTTP model also enters statistics in various scenario statistics database tables. Refer to *QualNet Statistics Database User's Guide* for details.

8.4.5.3 Dynamic Statistics

No dynamic statistics are supported for the HTTP model.

8.4.6 Sample Scenario

8.4.6.1 Scenario Description

There are HTTP servers on nodes 2, 5, 8, and 11. There is an HTTP client on node 1. This client chooses between servers 2, 5, and 11 when requesting web pages. It begins browsing 10 seconds after the start of simulation and remains idle for at most 2 minutes at a time.

8.4.6.2 Command Line Configuration

To configure the above scenario for the command line, include the following lines in the application configuration (.app) file.

```
HTTP 1 3 2 5 11 10S 120S
HTTPD 2
HTTPD 5
HTTPD 8
HTTPD 11
```

8.4.7 References

1. RFC1945, "Hypertext Transfer Protocol -- HTTP/1.0." T. Berners-Lee, R. Fielding, H. Frystyk. May 1996.
2. RFC2068, "Hypertext Transfer Protocol -- HTTP/1.1." R. Fielding, J. Gettys, J. Mogul, H. Frystyk, T. Berners-Lee. January 1997.
3. RFC2616, "Hypertext Transfer Protocol -- HTTP/1.1." R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee. June 1999.
4. Bruce Mah, "An Empirical Model of HTTP Network Traffic", INFOCOM 1997.

8.5 Lookup Traffic Generator

8.5.1 Description

Lookup is an unreliable query/response application that can be used to simulate applications such as DNS lookup or ping.

8.5.2 Command Line Configuration

To specify the Lookup application, include the following statement in the application configuration (.app) file:

```
LOOKUP <src> <dest> <num-requests-to-send> <request-size> <reply-size>
      <request-interval> <reply-delay> <start-time> <end-time>
      [TOS <tos-value> | DSCP <dscp-value> |
      PRECEDENCE <precedence-value>]
```

Note: All parameters should be entered on the same line.

Table 8-26 shows the Lookup application parameters. See [Section 1.2.1.1](#) for a description of the format used for the parameter table.

TABLE 8-26. Lookup Parameters

| Parameter | Value | Description |
|---|---|---|
| < <i>src</i> > <i>Required</i> | Integer or IP Address | Client node's ID or IP address. |
| < <i>dest</i> > <i>Required</i> | Integer or IP Address | Server node's ID or IP address. |
| < <i>num-request-to-send</i> > <i>Required</i> | Integer <i>Range:</i> ≥ 0 | Number of packets to send. If this is set to 0, items will be sent continually until < <i>end-time</i> > or until the end of the simulation, whichever comes first. See note 1. |
| < <i>request-size</i> > <i>Required</i> | Integer <i>Range:</i> [32 , 65023] <i>Unit:</i> bytes | Size of each request packet. |
| < <i>reply-size</i> > <i>Required</i> | Integer <i>Range:</i> [24 , 65023] <i>Unit:</i> bytes | Size of each reply size. |

TABLE 8-26. Lookup Parameters (Continued)

| Parameter | Value | Description |
|--|-----------------------------------|--|
| <request-interval> <i>Required</i> | Time <i>Range: > 0S</i> | Inter-departure interval between request packets. |
| <reply-delay> <i>Required</i> | Time <i>Range: > 0S</i> | Delay after receiving a request packet before responding with a reply packet. |
| <start-time> <i>Required</i> | Time <i>Range: > 0S</i> | Time when the transmission of packets should begin. |
| <end-time> <i>Required</i> | Time <i>Range: > 0S</i> | Time when the transmission of packets should end. If this is set to 0, transmission ends after <items-to-send> packets have been sent or until the end of simulation, whichever comes first. Note: <end-time> should be either greater than <start-time> or equal to 0. |
| TOS <tos-value> <i>Optional</i> | Integer <i>Range: [0, 255]</i> | Value of the 8-bit TOS field of the IP header for the packets generated. See note 2. |
| DSCH <dscp-value> <i>Optional</i> | Integer <i>Range: [0, 63]</i> | Value of the 6-bit DSCH field of the IP header for the packets generated. See note 2. |
| PRECEDENCE <precedence-value> <i>Optional</i> | Integer <i>Range: [0, 7]</i> | Value the 3-bit Precedence field of the IP header for the packets generated. See note 2. |

- Notes:**
- If <num-requests-to-send> and <end-time> are both greater than 0, Lookup will run until either <items-to-send> is done, <end-time> is reached, or the simulation ends, whichever comes first.
 - At most one of the three parameters PRECEDENCE, DSCH, and TOS can be specified. If PRECEDENCE, DSCH or TOS is not specified, PRECEDENCE 0 is used as default.

Scenario Configuration File Parameters

[Table 8-27](#) describes the Lookup Traffic Generator parameters that can be specified in the scenario configuration (.config) file.

TABLE 8-27. Lookup Traffic Generator Scenario Configuration File Parameters

| Parameter | Value | Description |
|--|--|---|
| APPLICATION-STATISTICS <i>Optional</i> Scope: Global, Node | List: • YES • NO <i>Default:</i> NO | Indicates whether statistics are collected for application protocols. |

Examples of Parameter Usage

The following are examples of Lookup configuration:

1. Node 1 sends to node 2 10 requests of 64bytes each every 1 second. Node 2 will reply to node 1 immediately after receiving a request packet from node 1. The size of each reply packet is 512bytes. Lookup starts at 0 seconds into simulation and stops after 10 seconds into simulation.

```
LOOKUP 1 2 10 64 512 1S 0S 0S 10S
```

2. Node 1 sends to node 2 10 requests of 64 bytes each every 1 second. Node 2 will reply to node 1 after 1 millisecond once a request packet is received from node 1. The size of each reply packet is 512 bytes. Lookup starts at 10 seconds into simulation and stops after 20 seconds into simulation.

```
LOOKUP 1 2 10 64 512 1S 1MS 10S 20S
```

8.5.3 GUI Configuration

Lookup Traffic Generator is configured using the Lookup Properties Editor.

To configure a Lookup session between two nodes, perform the following steps:

1. Click the **LOOK UP** button in the **Applications** tab of the Standard Toolset.
2. On the canvas, click on the source node, drag the mouse to the destination node, and release.
3. Open the Lookup Properties Editor by doing one of the following:
 - Right-click in the application link on the canvas and select **Properties** from the menu.
 - In the **Applications** tab of Table View, either double-click on the application row or right-click on the application row and select Properties from the menu.
4. Set the parameters listed in [Table 8-11](#).

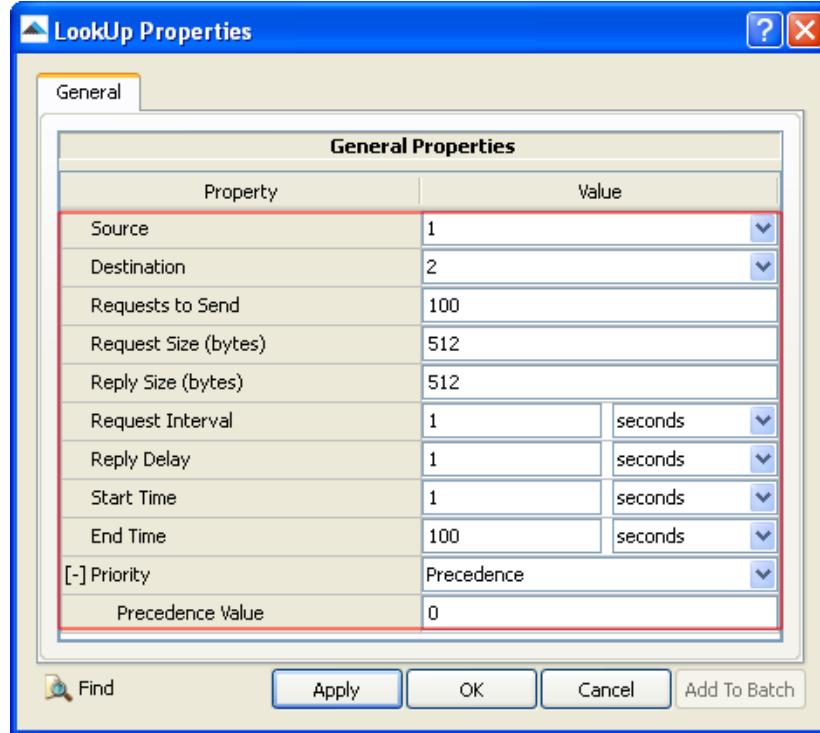


FIGURE 8-10. Setting Lookup Parameters

TABLE 8-28. GUI Parameters for Lookup

| GUI Parameter | Command Line Parameter |
|------------------------------|------------------------|
| Source | <src> |
| Destination | <dest> |
| Requests to Send | <num-request-to-send> |
| Request Size | <request-size> |
| Reply Size | <reply-size> |
| Request Interval | <request-interval> |
| Reply Delay | <reply-delay> |
| Start Time | <start-time> |
| End Time | <end-time> |
| Priority (set to DSCP) | DSCP |
| Priority (set to Precedence) | PRECEDENCE |
| Priority (set to TOS) | TOS |

Setting Parameters

- To specify an IP address as the source (destination) ID, set **Source (Destination)** to one of the IP addresses listed in the drop-down list.
- To set DSCP, Precedence, or TOS bits for packets, set **Priority** to *DSCP*, *Precedence*, or *TOS*, respectively.

5. If **Priority** is set to *DSCP*, *Precedence*, or *TOS*, set the appropriate dependent parameter listed in Table 8-29. Figure 8-11 shows how to set the dependent parameter when **Priority** is set to *DSCP*. Setting dependent parameters for the other two options is similar.

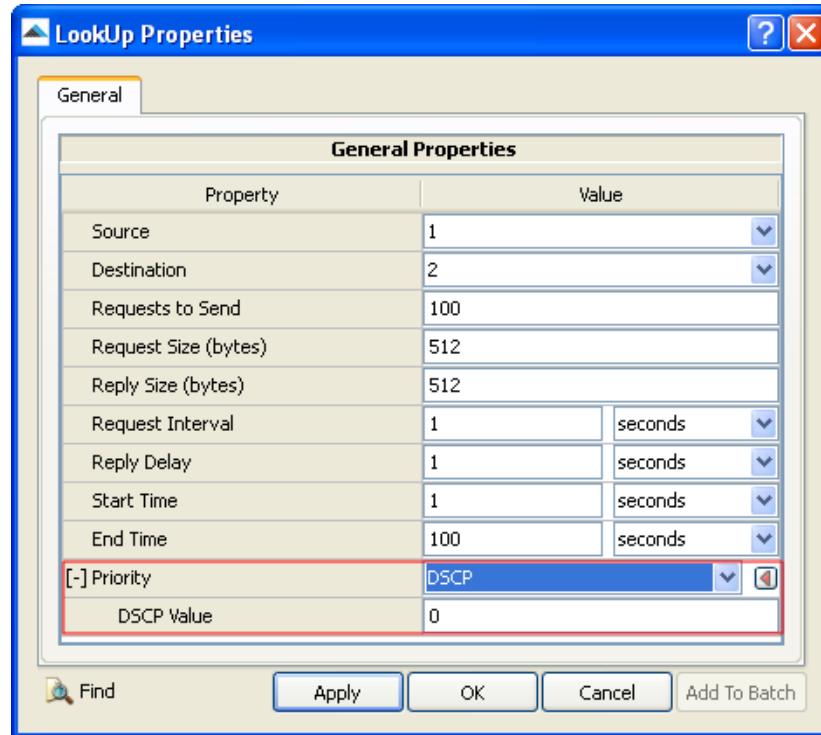


FIGURE 8-11. Setting DSCP Bits

TABLE 8-29. GUI Parameters for Specifying QoS for LOOKUP

| GUI Parameter | Command Line Parameter |
|------------------|------------------------|
| DSCP Value | <dscp-value> |
| Precedence Value | <precedence-value> |
| TOS Value | <tos-value> |

Configuring Statistics Parameters

Statistics for applications (including Lookup) can be collected at the global and node levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for Lookup, check the box labeled **Application** in the appropriate properties editor.

TABLE 8-30. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|----------------------|-------------------------------|-------------------------------|
| Application | Global, Node | APPLICATION-STATISTICS |

8.5.4 Statistics

[Table 8-31](#) list the Lookup statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 8-31. Lookup Statistics

| Statistic | Description |
|------------------------------|--|
| Lookup Client | |
| Server address | Address of the server. |
| First Packet Sent at (s) | Time in second when first packet was sent. |
| Last Packet Sent at (s) | Time in second when last packet was sent. |
| Session status | Current status of the session. |
| Total bytes Sent | Total number of bytes sent. |
| Total packets Sent | Total number of packets sent. |
| Total bytes received | Total number of bytes received. |
| Total packets received | Total number of packets received. |
| Average Roundtrip Delay (s) | Average value of round-trip delay (seconds). |
| Lookup Server | |
| Client address | Address of the client. |
| First packet received at (s) | Time in second when first packet was received. |
| Last packet received at (s) | Time in second when last packet was received. |
| Session status | Current status of the session. |
| Total Bytes Received | Total number of bytes received. |
| Total Packets Received | Total number of packets received. |
| Total bytes sent | Total number of bytes sent. |
| Total packets sent | Total number of packets sent. |

8.6 Multicast Constant Bit Rate (MCBR) Traffic Generator

8.6.1 Description

MCBR is used where there is an inherent reliance on time synchronization between the traffic source and destination multicast group. MCBR is tailored for any type of data for which the end-systems require predictable response time and a static amount of bandwidth continuously available for the life-time of the connection. These applications include services such as video conference, telephony (voice services) or any type of on-demand service, such as interactive voice and audio. For telephony and native voice applications, MCBR provides low-latency traffic with predictable delivery characteristics, and is typically used for circuit emulation. MCBR is useful where there are concerns about maintaining maximum compatibility, especially with certain streaming applications and some hardware-based decoders that don't reliably support variable bit rate.

8.6.2 Command Line Configuration

Configuration Requirements

A multicast protocol must be configured in order to use MCBR.

Application Configuration File Parameters

To specify MCBR traffic, include the following statement in the application configuration (.app) file:

```
MCBR <src> <multicast-destination> <items-to-send> <item-size>
      <interval> <start-time> <end-time>
      [TOS <tos-value> | DSCP <dscp-value> |
       PRECEDENCE <precedence-value>]
      [MDP-ENABLED [MDP-PROFILE <profile-name>] ]
      [APPLICATION-NAME <application-name>]
```

Note: All parameters should be entered on the same line.

The MCBR parameters are described in [Table 8-32](#). See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 8-32. MCBR Parameters

| Parameter | Value | Description |
|--|-----------------------------------|--|
| <src> <i>Required</i> | Integer or IP Address | Client node's ID or IP address. |
| <multicast-destination> <i>Required</i> | IP Address | Group multicast address to which data packets are sent. |
| <items-to-send> <i>Required</i> | Integer <i>Range:</i> ≥ 0 | Number of packets to send. If this is set to 0, items will be sent continually until <end-time> or until the end of the simulation, whichever comes first. See note 1. |

TABLE 8-32. MCBR Parameters (Continued)

| Parameter | Value | Description |
|--|---|---|
| <item-size> <i>Required</i> | Integer <i>Range:</i> [13 , 65203] <i>Unit:</i> bytes | Size of each item. |
| <interval> <i>Required</i> | Time <i>Range:</i> > 0S | Time between transmission of successive packets (inter-departure time). |
| <start-time> <i>Required</i> | Time <i>Range:</i> ≥ 0S | Time when the transmission of packets should begin. |
| <end-time> <i>Required</i> | Time <i>Range:</i> ≥ 0S | Time when the transmission of packets should end. If this is set to 0, transmission ends after <items-to-send> packets have been sent or until the end of simulation, whichever comes first. <end-time> should be either greater than <start-time> or equal to 0. |
| TOS <tos-value> <i>Optional</i> | Integer <i>Range:</i> [0 , 255] | Value of the 8-bit TOS field of the IP header for the packets generated. See note 3. |
| DSCP <dscp-value> <i>Optional</i> | Integer <i>Range:</i> [0 , 63] | Value of the 6-bit DSCP field of the IP header for the packets generated. See note 3. |
| PRECEDENCE <precedence-value> <i>Optional</i> | Integer <i>Range:</i> [0 , 7] | Value of the 3-bit Precedence field of the IP header for the packets generated. See note 3. |
| MDP-ENABLED <i>Optional</i> | N/A | Keyword which specifies that MDP is enabled for the application. Note: If this keyword is not included, then the application does not run with MDP. |

TABLE 8-32. MCBR Parameters (Continued)

| Parameter | Value | Description |
|--|--------|---|
| MDP-PROFILE <i><profile-name></i> <i>Optional</i> | String | <p>Name of the MDP profile to be used with the application.</p> <p>This should be the name of a MDP profile defined in the MDP profile file (see Section 7.2.4.3).</p> <p>This MDP profile takes precedence over the MDP profile associated with the node in the scenario configuration file, if any.</p> <p>Note: This parameter can be included only if the parameter MDP-ENABLED is also included.</p> <p>Note: If parameter MDP-ENABLED is included but parameter MDP-PROFILE is not included, and a MDP profile is specified for the node in the scenario configuration file, then the MDP profile specified for the node is used.</p> <p>If parameter MDP-ENABLED is included but parameter MDP-PROFILE is not included, and a MDP profile is not specified for the node in the scenario configuration file, then default MDP values for the MDP profile parameters are used (see Section 7.2.4.3).</p> |
| APPLICATION-NAME <i><application-name></i> <i>Optional</i> | String | Name of the MCBR session. This name is printed in the statistics file and statistics database. |

- Note:**
1. If *<end-time>* is set to 0, MCBR runs until all *<items-to-send>* are transmitted, or until the end of simulation, whichever comes first.
 2. If *<items-to-send>* and *<end-time>* are both greater than 0, MCBR runs until either *<items-to-send>* is done, *<end-time>* is reached, or the simulation ends, whichever comes first.
 3. At most one of the three parameters PRECEDENCE, DSCP, and TOS can be specified. If PRECEDENCE, DSCP, and TOS are not specified, PRECEDENCE 0 is used as default.

Scenario Configuration File Parameters

[Table 8-33](#) describes the MCBR parameters that can be specified in the scenario configuration (.config) file.

TABLE 8-33. MCBR Scenario Configuration File Parameters

| Parameter | Value | Description |
|--|--|--|
| APPLICATION-STATISTICS <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Indicates whether statistics are collected for application protocols (including MCBR). |

Examples of Parameter Usage

The following are examples of MCBR configuration:

1. Node 1 sends to IPv4 multicast address 225.0.0.1 ten items of 1460 bytes each at the start of the simulation, and up to 600 seconds into the simulation. The inter-departure time for each item is 1 second.

```
MCBR 1 225.0.0.1 10 1460 1S 0S 600S
```

2. Node 1 continuously sends to IPv4 multicast address 225.0.0.1 items of 1460 bytes each at the start of the simulation up to 600 seconds into the simulation. The inter-departure time for each item is 1 second.

```
MCBR 1 225.0.0.1 0 1460 1S 0S 600S
```

3. Node 1 continuously sends to IPv4 multicast address 225.0.0.1 items of 1460 bytes each at the start of the simulation up to the end of the simulation. The inter-departure time for each item is 1 second.

```
MCBR 1 225.0.0.1 0 1460 1S 0S 0S
```

4. Node 1 continuously sends to IPv6 multicast address ff1E items of 1460 bytes each at the start of the simulation up to the end of the simulation. The inter-departure time for each item is 1 second.

```
MCBR 1 ff1E::3 0 1460 1S 0S 0S
```

5. This is the same as the previous example, except that MCBR runs with MDP and uses the MDP profile associated with node 1. If there is no MDP profile associated with node 1, default values for MDP profile parameters are used.

```
MCBR 1 ff1E::3 0 1460 1S 0S 0S MDP-ENABLED
```

6. This is the same as the previous example, except that MDP uses the user-defined MDP profile profile-1.

```
MCBR 1 ff1E::3 0 1460 1S 0S 0S MDP-ENABLED MDP-PROFILE profile-1
```

8.6.3 GUI Configuration

MCBR is configured using the MCBR Properties Editor.

To configure a MCBR Single Host Application, perform the following steps:

1. Click the **MCBR** button in the **Single Host Applications** tab of the Standard Toolset.
2. On the canvas, click on the source node.
3. Open the MCBR Properties Editor by either double-clicking on the application row in the **Applications** tab of Table View or right-clicking on the application row and selecting **Properties** from the menu.

4. Set the parameters listed in Table 8-34.

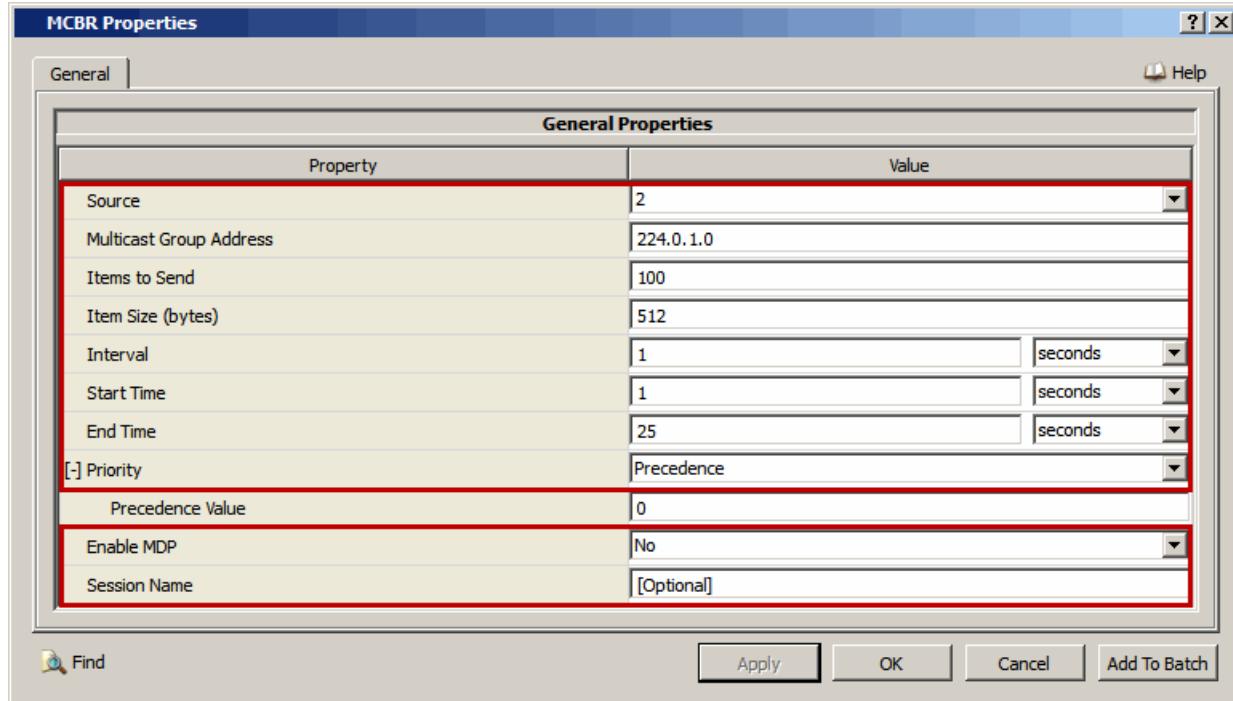


FIGURE 8-12. Setting MCBR Parameters

TABLE 8-34. Command Line Equivalent of MCBR Parameters

| GUI Parameter | Command Line Parameter |
|------------------------------|-------------------------------------|
| Source | <src> |
| Multicast Group Address | <multicast-destination> |
| Items to Send | <items-to-send> |
| Item Size | <item-size> |
| Interval | <interval> |
| Start Time | <start-time> |
| End Time | <end-time> |
| Priority (set to DSCP) | DSCP |
| Priority (set to Precedence) | PRECEDENCE |
| Priority (set to TOS) | TOS |
| Enable MDP (set to Yes) | MDP-ENABLED |
| Session Name | APPLICATION-NAME <application-name> |

Setting Parameters

- To specify an IP address as the source ID, set **Source** to one of the IP addresses listed in the drop-down list.

- Set **Multicast Group Address** to the address of the multicast group that is to receive traffic from the source.
 - To set DSCP, Precedence, or TOS bits for packets, set **Priority** to *DSCP*, *Precedence*, or *TOS*, respectively.
 - To enable MDP, set **Enable MDP** to Yes.
5. If **Priority** is set to *DSCP*, *Precedence*, or *TOS*, set the appropriate dependent parameter listed in [Table 8-35](#). [Figure 8-13](#) shows how to set the dependent parameter when **Priority** is set to *DSCP*. Setting dependent parameters for the other two options is similar.

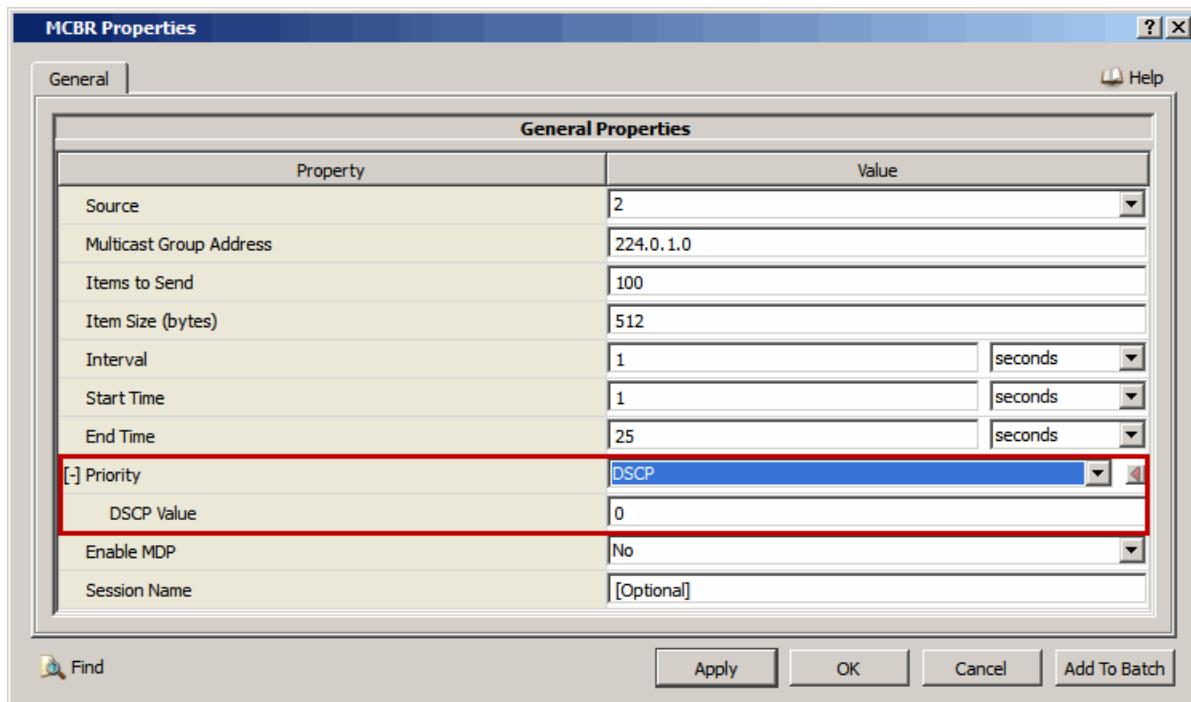


FIGURE 8-13. Setting DSCP Value

TABLE 8-35. Command Line Equivalent of Priority Parameters

| GUI Parameter | Command Line Parameter |
|------------------|------------------------|
| DSCP Value | <dscp-value> |
| Precedence Value | <precedence-value> |
| TOS Value | <tos-value> |

6. If **Enable MDP** is set to Yes, then set the parameters listed in [Table 8-36](#).

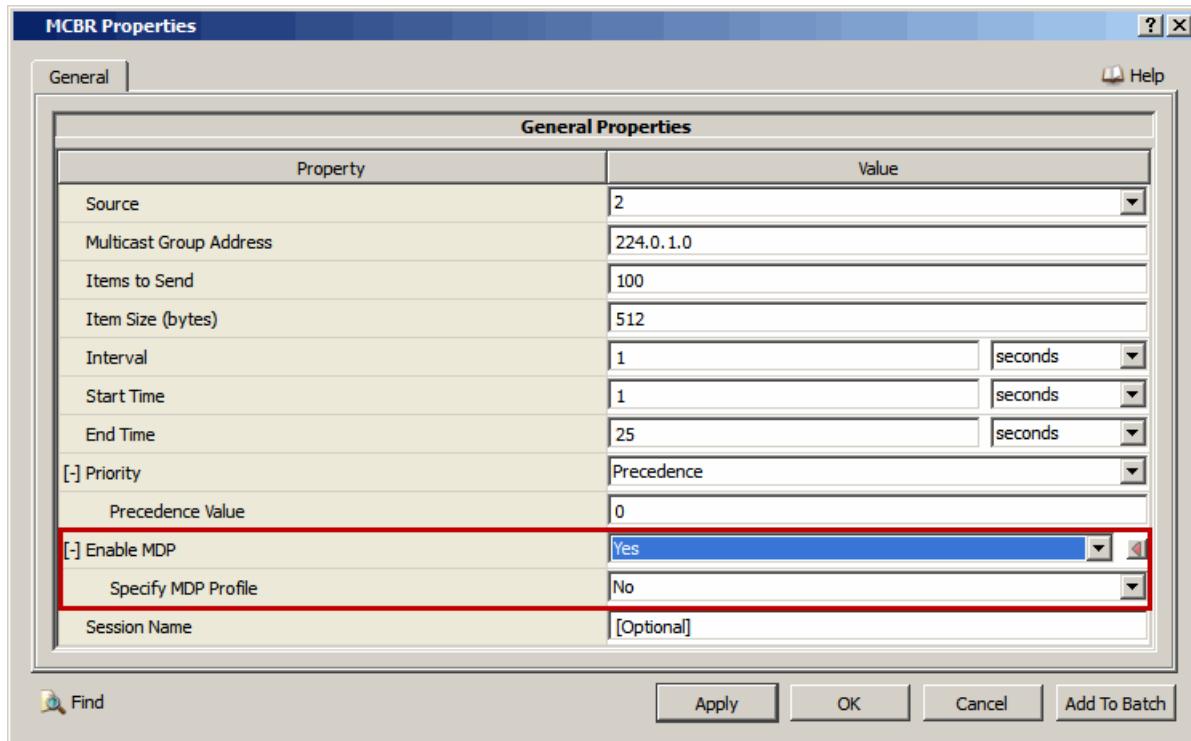


FIGURE 8-14. Enabling MDP

TABLE 8-36. Command Line Equivalent of MDP Parameters

| GUI Parameter | Command Line Parameter |
|----------------------------------|------------------------|
| Specify MDP Profile (set to Yes) | MDP - PROFILE |

Setting Parameters

- To specify an MDP profile, set **Specify MDP Profile** to Yes. To use the default MDP profile, set **Specify MDP Profile** to No.

7. If **Specify MDP Profile** is set to Yes, then set the parameters listed in [Table 8-37](#).

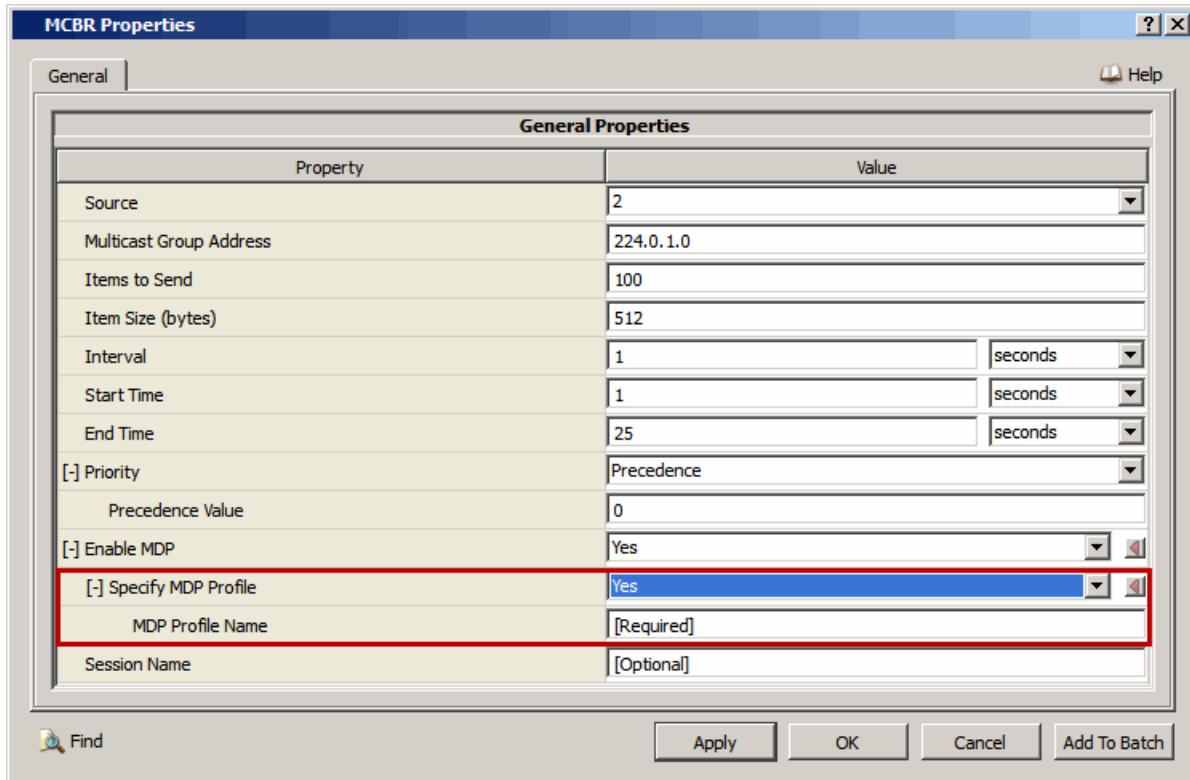


FIGURE 8-15. Specifying MDP Profile

TABLE 8-37. Command Line Equivalent of MDP Profile Parameters

| GUI Parameter | Command Line Parameter |
|------------------|------------------------|
| MDP Profile Name | <profile-name> |

Configuring Statistics Parameters

Statistics for applications (including MCBR) can be collected at the global and node levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for MCBR, check the box labeled **Application** in the appropriate properties editor.

TABLE 8-38. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| Application | Global, Node | APPLICATION-STATISTICS |

8.6.4 Statistics

This section describes the file, database, and dynamic statistics of the MCBR model.

8.6.4.1 File Statistics

[Table 8-39](#) lists the MCBR statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 8-39. MCBR Statistics

| Statistic | Description |
|--|--|
| MCBR Client Statistics | |
| Server Address | Specifies the IP address of the server. |
| Session Status | Status of the session (open or closed) at the end of simulation. |
| Multicast Session Start (seconds) | Time in seconds, when multicast session was started. |
| Multicast Session Finish (seconds) | Time in seconds, when multicast session was finished. |
| First Multicast Fragment Sent (seconds) | Time in seconds, when first multicast fragment was sent. |
| Last Multicast Fragment Sent (seconds) | Time in seconds, when last multicast fragment was sent. |
| Total Multicast Fragments Sent (fragments) | Total number of multicast fragments sent. |
| First Multicast Message Sent (seconds) | Time in seconds, when first multicast message was sent. |
| Last Multicast Message Sent (seconds) | Time in seconds, when last multicast message was sent. |
| Total Multicast Messages Sent (messages) | Total number of multicast messages sent. |
| Total Multicast Data Sent (bytes) | Total number of multicast data bytes sent. |
| Total Multicast Overhead Sent (bytes) | Total number of multicast overhead bytes sent. |
| Multicast Offered Load (bits/second) | Total multicast offered load. |
| MCBR Server Statistics | |
| Client Address | Specifies the IP address of the client. |
| Session Status | Specifies whether the session status is open or closed. |
| Multicast Session Start (seconds) | Time in seconds, when multicast session was started. |
| Multicast Session Finish (seconds) | Time in seconds, when multicast session was finished. |
| First Multicast Fragment Received (seconds) | Time in seconds, when first multicast fragment was received. |
| Last Multicast Fragment Received (seconds) | Time in seconds, when last multicast fragment was received. |
| Total Multicast Fragments Received (fragments) | Total number of multicast fragments received. |
| First Multicast Message Received (seconds) | Time in seconds, when first multicast message was received. |
| Last Multicast Message Received (seconds) | Time in seconds, when last multicast message was received. |
| Total Multicast Messages Received (messages) | Total number of multicast messages received. |
| Total Multicast Data Received (bytes) | Total number of multicast data bytes received. |
| Total Multicast Overhead Received (bytes) | Total number of multicast overhead bytes received. |
| Average Multicast End-to-End Delay (seconds) | Average multicast end-to-end delay. |
| Multicast Received Throughput (bits/second) | Multicast received throughput at the server. |
| Average Multicast Jitter (seconds) | Average multicast jitter. |

Notes: 1. The average end-to-end delay at the server is calculated as follows:

(Total of packet delays for all packets) / (Total packets received)

where packet delay = (time when packet is received at the server - time when the packet is transmitted at the client)

2. The throughput at the server is calculated as follows:

- If the session is complete, throughput = (total bytes received * 8) / (time last packet received - time first packet received), where the times are in seconds.
- If the session is incomplete, throughput = (total bytes received * 8) / (simulation time - time first packet received), where the times are in seconds.

8.6.4.2 Database Statistics

In addition to the file statistics, the MCBR model also enters statistics in various scenario statistics database tables. Refer to *QualNet Statistics Database User's Guide* for details.

8.6.4.3 Dynamic Statistics

No dynamic statistics are supported for the MCBR model.

8.7 Super Application Traffic Generator

8.7.1 Description

Super Application is a generic traffic generator. Super Application can simulate both UDP and TCP flows. Two-way flow (request-response) is supported for UDP based applications. Request packets travel from source to destination, whereas response packets travel from destination to source.

8.7.2 Limitations and Assumptions

- Responses are only supported with unreliable delivery type.

8.7.3 Command Line Configuration

The application parameters for the Super Application traffic generator have to be specified in the application configuration (.app) file (see [Section 8.7.3.1](#)). In addition, some Super Application related parameters have to be specified in the scenario configuration (.config) file (see [Section 8.7.3.2](#)).

8.7.3.1 Super Application Parameters Specified in Application Configuration File

To use Super Application, include the following statement in the application configuration (.app) file:

```
SUPER-APPLICATION <source>
    <destination>
        START-TIME <start-time>
        DURATION <duration>
        DELIVERY-TYPE <delivery-type>
        [CONNECTION-RETRY <retry-parameters>]
        REQUEST-NUM <request-num>
        [REQUEST-SIZE <request-size>]
        [REQUEST-INTERVAL <request-interval>]
        [REQUEST-DATA-RATE <data-rate>]
        [REQUEST-TOS <request-QoS>]
        [REPLY-PROCESS <reply-or-not>]
        [REPLY-NUM <reply-num>]
        [REPLY-SIZE <reply-size>]
        [REPLY-PROCESS-DELAY <reply-delay>]
        [REPLY-INTERDEPARTURE-DELAY <inter-reply-time>]
        [REPLY-TOS <reply-QoS>]
        [ENCODING-SCHEME <scheme-parameters>]
        [AVERAGE-TALK-TIME <avg-talk-time>]
        [MEAN-SPURT-GAP-RATIO <gap-ratio>]
        [FRAGMENT-SIZE <fragment-size>]
        [SOURCE-PORT <source-port>]
        [DESTINATION-PORT <destination-port>]
        [CHAIN-ID <chain-id>]
        [APPLICATION-NAME <application-name>]
        [MDP-ENABLED [MDP-PROFILE <profile-name>]]
        [CONDITIONS <conditions>]
        [REPEAT <repeat-interval> <no-of-repeats>]
```

Note: All parameters should be entered on the same line.

The Super Application parameters are described in Table 8-40. See Section 1.2.1.1 for a description of the format used for the parameter table.

TABLE 8-40. Super Application Parameters

| Parameter | Value | Description |
|--|--------------------------------|---|
| <source> <i>Required</i> | Integer or IP Address | Client node's ID or IP address. |
| <destination> <i>Required</i> | Integer or IP Address | Server node's ID or IP address. |
| START-TIME <start-time> <i>Required</i> | Time distribution (see note 1) | Note: Start time of the Super Application session. |

TABLE 8-40. Super Application Parameters (Continued)

| Parameter | Value | Description |
|---|--|---|
| DURATION <duration> <i>Required</i> | Time distribution (see note 1) | Duration of the Super Application session. Note: If the value of this parameter is 0, then the application runs till the end of simulation. |
| DELIVERY-TYPE <delivery-type> <i>Required</i> | List: • RELIABLE • UNRELIABLE | Indicates whether TCP (corresponding to option RELIABLE) or UDP (corresponding to option UNRELIABLE) is used as the transport protocol. |
| CONNECTION-RETRY <retry-parameters> <i>Optional</i> | Retry parameters specification (see note 2) <i>Default:</i> 0 DET 1S | TCP connection retry parameters. See note 2. Connection retry parameters must be specified if TCP is used as the transport protocol, i.e., if DELIVERY-TYPE RELIABLE is specified. |
| REQUEST-NUM <request-num> <i>Required</i> | Integer distribution (see note 1) | Number of request packets to send. |
| REQUEST-SIZE <request-size> <i>Optional</i> | Integer distribution (see note 1) <i>Unit:</i> bytes | Size of a request packet. Note: Exactly two of the following parameters must be specified: REQUEST-SIZE, REQUEST-INTERVAL, and REQUEST-DATA-RATE. |
| REQUEST-INTERVAL <request-interval> <i>Optional</i> | Time distribution (see note 1) | Delay between sending successive request packets. Note: This is the delay between successive packets, not between fragments of the same packet. If fragmentation occurs, the fragments are sent back-to-back. Note: Exactly two of the following parameters must be specified: REQUEST-SIZE, REQUEST-INTERVAL, and REQUEST-DATA-RATE. |
| REQUEST-DATA-RATE <data-rate> <i>Optional</i> | Real <i>Unit:</i> bytes/sec | Requested data rate. Note: Exactly two of the following parameters must be specified: REQUEST-SIZE, REQUEST-INTERVAL, and REQUEST-DATA-RATE. |
| REQUEST-TOS <request-QoS> <i>Optional</i> | QoS specification (see note 3) <i>Default:</i> TOS 0 | Type of service for request packets. |
| REPLY-PROCESS <reply-or-not> <i>Optional</i> | List: • YES • NO <i>Default:</i> NO | Specifies whether a response is sent for each request. Replies are not sent unless this parameter is set to YES. For the multicast mode, this parameter should be set to NO. Note: If this parameter is set to YES, then parameters REPLY-NUM, REPLY-SIZE, REPLY-PROCESS-DELAY, and REPLY-INTERDEPARTURE-DELAY must be specified. |

TABLE 8-40. Super Application Parameters (Continued)

| Parameter | Value | Description |
|--|--|--|
| REPLY-NUM <reply-num> <i>Optional</i> | Integer distribution (see note 1) | Number of reply packets sent in response to each request packet. |
| REPLY-SIZE <reply-size> <i>Optional</i> | Integer distribution (see note 1) <i>Unit: bytes</i> | Size of a reply packet. |
| REPLY-PROCESS-DELAY <reply-delay> <i>Optional</i> | Time distribution (see note 1) | Delay between the time when the destination receives a request and the time when it sends the first response packet in response to the request. |
| REPLY-INTERDEPARTURE-DELAY <inter-reply-time> <i>Optional</i> | Time distribution (see note 1) | Delay between sending successive response packets in response to the same request. Note: This is the delay between successive packets, not between fragments of the same packet. If fragmentation occurs, the fragments are sent back-to-back. |
| REPLY-TOS <reply-QoS> <i>Optional</i> | QoS specification (see note 3) <i>Default: TOS 0</i> | Type of service for reply packets. |
| ENCODING-SCHEME <scheme-parameters> <i>Optional</i> | Encoding scheme parameter specification (see note 4) | Encoding scheme to be used. Encoding scheme must be specified if audio or video is transmitted. The specification consists of the traffic type (voice or video) and the codec to use (see note 4). Note: If the encoding scheme is specified, then UDP is used, irrespective of the DELIVERY-TYPE parameter, and default values for packet size and packet interval for the specified codec are used, irrespective of the REQUEST-INTERVAL and REQUEST-SIZE parameters. Note: If the encoding scheme is specified, then the parameter REPLY-PROCESS should not be included. Note: If the traffic type is specified as VOICE (see note 4), then parameters AVERAGE-TALK-TIME and MEAN-SPURT-GAP-RATIO must be specified. |
| AVERAGE-TALK-TIME <avg-talk-time> <i>Optional</i> | Time distribution (see note 1) | Average talk time of the voice session. Note: This parameter must be specified if the traffic type is specified as VOICE (see note 4). |

TABLE 8-40. Super Application Parameters (Continued)

| Parameter | Value | Description |
|--|-----------------------------------|--|
| MEAN-SPURT-GAP-RATIO <i><gap-ratio></i> <i>Optional</i> | Integer | Gap ratio between talk spurts. AVERAGE-TALK-TIME and MEAN-SPURT-GAP-RATIO are used to calculate the talk time. The session initiator will transmit the data/voice until talk time expires. Note: This parameter must be specified if the traffic type is specified as VOICE (see note 4). |
| FRAGMENT-SIZE <i><fragment-size></i> <i>Optional</i> | Integer <i>Unit: bytes</i> | Maximum size of a packet fragment. If a packet is larger than this parameter, then it is broken into multiple fragments, each (except possibly the last) of size FRAGMENT-SIZE. Note: If a packet is fragmented, then all fragments of a packet are sent back to back. |
| SOURCE-PORT <i><source-port></i> <i>Optional</i> | Integer | Port number to be used at the source. The application chooses a free source port when the user does not explicitly specify a source port. User-specified ports must be unique. |
| DESTINATION-PORT <i><destination-port></i> <i>Optional</i> | Integer | Port number to be used at the destination. The application uses its default port when a destination port is not specified by the user. User-specified ports must be unique. |
| CHAIN-ID <i><chain-id></i> <i>Optional</i> | String | Chain ID of the super application session. |
| APPLICATION-NAME <i><application-name></i> <i>Optional</i> | String | Name of the Super Application session. This name is printed in the statistics file and statistics database. |
| MDP-ENABLED <i>Optional</i> | N/A | Keyword which specifies that MDP is enabled for the application. Note: If this keyword is not included, then the application does not run with MDP. |

TABLE 8-40. Super Application Parameters (Continued)

| Parameter | Value | Description |
|--|---------------------------------------|---|
| MDP-PROFILE <i><profile-name></i> <i>Optional</i> | String | <p>Name of the MDP profile to be used with the application.</p> <p>This should be the name of a MDP profile defined in the MDP profile file (see Section 7.2.4.3).</p> <p>This MDP profile takes precedence over the MDP profile associated with the node in the scenario configuration file, if any.</p> <p>Note: This parameter can be included only if the parameter MDP-ENABLED is also included.</p> <p>Note: If parameter MDP-ENABLED is included but parameter MDP-PROFILE is not included, and a MDP profile is specified for the node in the scenario configuration file, then the MDP profile specified for the node is used.</p> <p>If parameter MDP-ENABLED is included but parameter MDP-PROFILE is not included, and a MDP profile is not specified for the node in the scenario configuration file, then default MDP values for the MDP profile parameters are used (see Section 7.2.4.3).</p> |
| CONDITIONS <i><conditions></i> <i>Optional</i> | Conditions specification (see note 5) | <p>Conditions to determine when to trigger the Super Application session.</p> <p>Both TCP and UDP sessions support trigger conditions.</p> <p>Conditions are specified by means of a Boolean expression (see note 5).</p> |
| REPEAT <i><repeat-interval></i> <i><no-of-repeats></i> <i>Optional</i> | See description | <p>Repetition parameters for the entire application flow.</p> <p><i><repeat-interval></i> Time to wait after a flow ends before repeating the entire flow. This is specified as a time distribution (see note 1).</p> <p><i><no-of-repeats></i> Number of times the entire flow is repeated. This is specified as an integer distribution (see note 1).</p> |

Notes: 1. **Integer and Time Distributions:** Several Super Application parameters are specified as random number distributions. Three random number distributions are supported: deterministic, uniform, and exponential.

- The deterministic distribution is specified as:

```
DET <value>
```

It always returns `<value>` as the value.

- The uniform distribution is specified as:

```
UNI <value-1> <value-2>
```

It returns a value uniformly distributed between `<value-1>` and `<value-2>`.

- The exponential distribution is specified as:

```
EXP <value>
```

It returns a value from an exponential distribution with `<value>` as the mean.

For integer distributions, `<value>`, `<value-1>`, and `<value-2>` are integer values, e.g., 0, 10, 15, etc.

For time distributions, `<value>`, `<value-1>`, and `<value-2>` are time values, e.g., 5S, 0.5MS, 100US, etc.

2. Retry Parameters Specification:

Retry parameters specification has the following format:

```
<max-retries> <retry-interval>
```

where

| | |
|----------------------------------|---|
| <code><max-retries></code> | Maximum number of attempts to establish a TCP connection. This is an integer value (≥ 0). 0 indicates that there is no maximum, i.e., TCP connection establishment is tried continually until a connection is established or the end of simulation is reached. |
|----------------------------------|---|

The default value is 0.

| | |
|-------------------------------------|---|
| <code><retry-interval></code> | Time between two consecutive attempts to establish a TCP connection. This is specified as a time distribution (see note 1). |
|-------------------------------------|---|

The default value is DET 1S.

3. QoS Specification: Type of service for request and reply packets can be specified by including a TOS specification, DSCP specification, or Precedence specification.

- TOS specification has the following format:

TOS <TOS-value>

where <TOS-value> is the value of the TOS bits of the IP header.

- DSCP specification has the following format:

DSCP <DSCP-value>

where <DSCP-value> is the value of the DSCP bits of the IP header.

- Precedence specification has the following format:

PRECEDENCE <precedence-value>

where <precedence-value> is the value of the Precedence bits of the IP header.

If a TOS specification, DSCP specification, or Precedence specification is not included, then PRECEDENCE 0 is used as default.

4. Encoding Scheme Parameter Specification: Encoding scheme parameter specification has the following format:

```
<traffic-type> <codec>
```

where

| | |
|-----------------------------|--|
| <traffic-type> | Type of traffic. This can be can be VIDEO or VOICE |
| <codec> | Codec to use. |

If <traffic-type> is VIDEO, then <codec> can be one of the following: H.261, H.263, MPEG1.M, MPEG1.H, MPEG2.M, MPEG2.H

If <traffic-type> is VOICE, then <codec> can be one of the following: G.711, G.729, G.723.lar6.3, G.723.lar5.3, G.726ar32, G.726ar24, G.728ar16, CELP, MELP

The default packet size and packet interval for each codec type are listed below.

| Codec | Default Packet Size | Default Packet Interval |
|--------------|----------------------------|--------------------------------|
| | (bytes) | (milliseconds) |
| H.261 | 160 | 20 |
| H.263 | 160 | 20 |
| MPEG1.M | 2500 | 20 |
| MPEG1.H | 7500 | 20 |
| MPEG2.M | 12500 | 20 |
| MPEG2.H | 37500 | 20 |
| G.711 | 160 | 20 |
| G.729 | 20 | 20 |
| G.723.lar6.3 | 23 | 30 |
| G.726ar32 | 80 | 20 |
| G.726ar24 | 60 | 20 |
| G.728ar16 | 40 | 30 |
| CELP | 18 | 30 |
| MELP | 8 | 22.5 |

5. Conditions Parameter Specification: Conditions parameter specification has the following format:

```
CHAIN-ID = <chain-id> AND PACKET <op-type> <num-packets>
```

where

| | |
|---------------|--|
| <chain-id> | Chain ID, specified as a string value. |
| <op-type> | Operator type, which can be one of the following: >=, <=, >, <, = |
| <num-packets> | Number of packets, specified as an integer value. |

Complex boolean expressions (built recursively using the simple expression and the boolean operators AND, OR, XOR, NOT, and parentheses) can also be used as the conditions parameter specification. The recursion rule is summarized by the following:

```
<expr> ::= <expr> AND <expr> |
           <expr> OR <expr> |
           <expr> XOR <expr> |
           ( <expr> ) |
           NOT <expr> |
           <condition>
```

where <condition> is the simple condition expression described above.

8.7.3.2 Super Application Parameters Specified in the Scenario Configuration File

Table 8-41 describes the Super Application parameters that can be specified in the scenario configuration (.config) file.

TABLE 8-41. Super Application Scenario Configuration File Parameters

| Parameter | Value | Description |
|--|--|---|
| APPLICATION-STATISTICS <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Indicates whether statistics are collected for application protocols (including Super Application). |
| TRACE-SUPERAPPLICATION <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> YES | Indicates whether packet tracing is enabled for Super Application. Note: To enable packet tracing, some other parameters need to be configured as well. Refer to Section 4.2.10 of <i>QualNet User's Guide</i> for details. |

Examples of Parameter Usage

The following are examples of Super Application configuration:

1. Node 1 sends to node 2 five items of 1460 bytes each using TCP from 60 seconds simulation time till 70 seconds simulation time. If five items are sent before 10 seconds elapsed, no other items are sent. Precedence is set to 0.

```
SUPER-APPLICATION 1 2 DELIVERY-TYPE RELIABLE START-TIME DET 60S
DURATION DET 10S REQUEST-NUM DET 5
REQUEST-SIZE DET 1460 REQUEST-INTERVAL DET 0S
REQUEST-TOS PRECEDENCE 0 REPLY-PROCESS NO
```

2. Node 1 sends to node 2 three (request) packets of 123 bytes each using UDP. Node 1 starts sending at 10 seconds into the simulation and sends for the duration of 21 seconds. Packets are sent with a 1 second interval with precedence set to 1. Node 1 uses port 2345 whereas node 2 uses port 1751. The fragment size is bigger than packet size, so no fragmentation takes place. On receiving the request packets, node 2 replies with 2 packets of 140 bytes each, for every request packet received. On receiving a request packet, node 2 waits 1 second before sending the first reply packet for that request packet. The second reply packet is sent 0.1 seconds after the first one.

```
SUPER-APPLICATION 1 2 DELIVERY-TYPE UNRELIABLE START-TIME DET 10S
DURATION DET 21S REQUEST-NUM DET 3
REQUEST-SIZE DET 123 REQUEST-INTERVAL DET 1S REQUEST-
TOS PRECEDENCE 1 REPLY-PROCESS YES FRAGMENT-SIZE 200
DESTINATION-PORT 1751 SOURCE-PORT 2345
REPLY-NUM DET 2 REPLY-SIZE DET 140
REPLY-PROCESS-DELAY DET 1S
REPLY-INTERDEPARTURE-DELAY DET 0.1S
```

3. This is the same as the previous example, except that Super Application runs with MDP and uses the MDP profile associated with node 1. If there is no MDP profile associated with node 1, default values for MDP profile parameters are used.

```
SUPER-APPLICATION 1 2 DELIVERY-TYPE UNRELIABLE START-TIME DET 10S
DURATION DET 21S REQUEST-NUM DET 3
REQUEST-SIZE DET 123 REQUEST-INTERVAL DET 1S REQUEST-
TOS PRECEDENCE 1 REPLY-PROCESS YES FRAGMENT-SIZE 200
DESTINATION-PORT 1751 SOURCE-PORT 2345
REPLY-NUM DET 2 REPLY-SIZE DET 140
REPLY-PROCESS-DELAY DET 1S
REPLY-INTERDEPARTURE-DELAY DET 0.1S
MDP-ENABLED
```

4. This is the same as the previous example, except that MDP uses the user-defined MDP profile profile-1.

```
SUPER-APPLICATION 1 2 DELIVERY-TYPE UNRELIABLE START-TIME DET 10S
DURATION DET 21S REQUEST-NUM DET 3
REQUEST-SIZE DET 123 REQUEST-INTERVAL DET 1S REQUEST-
TOS PRECEDENCE 1 REPLY-PROCESS YES FRAGMENT-SIZE 200
DESTINATION-PORT 1751 SOURCE-PORT 2345
REPLY-NUM DET 2 REPLY-SIZE DET 140
```

```
REPLY-PROCESS-DELAY DET 1S
REPLY-INTERDEPARTURE-DELAY DET 0.1S
MDP-ENABLED MDP-PROFILE profile-1
```

8.7.4 GUI Configuration

Section 8.7.4.1 describes how to configure a Super Application session between two nodes.

Section 8.7.4.1 describes how to configure a single host Super Application session. Section 8.7.4.1 describes how to configure statistics parameters for applications (including Super Application).

8.7.4.1 Configuring a Client-Server Super Application Session

To configure a Super Application session between two nodes, perform the following steps:

1. Click the **SUPR APP** button in the **Applications** tab of the Standard Toolset.
2. On the canvas, click on the source node, drag the mouse to the destination node, and release.
3. Open the Super Application Properties Editor by doing one of the following:
 - a. Right-click in the application link on the canvas and select **Properties** from the menu.
 - b. In the **Applications** tab of Table View either double-click on the application row or right-click on the application row and select **Properties** from the menu.

4. Set the parameters listed in Table 8-42.

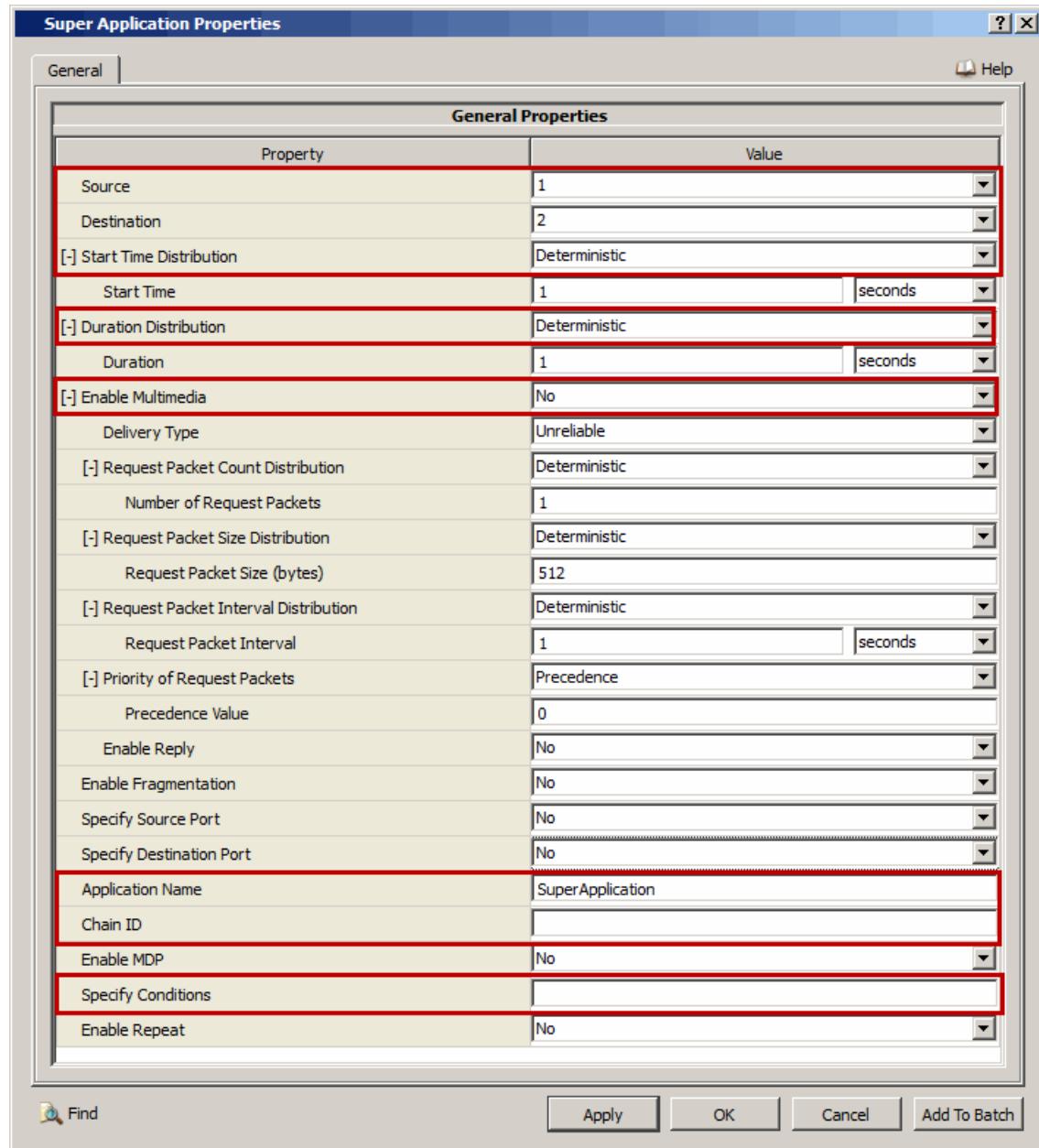


FIGURE 8-16. Setting Super Application Parameters

TABLE 8-42. Command Line Equivalent of Supper Application Parameters

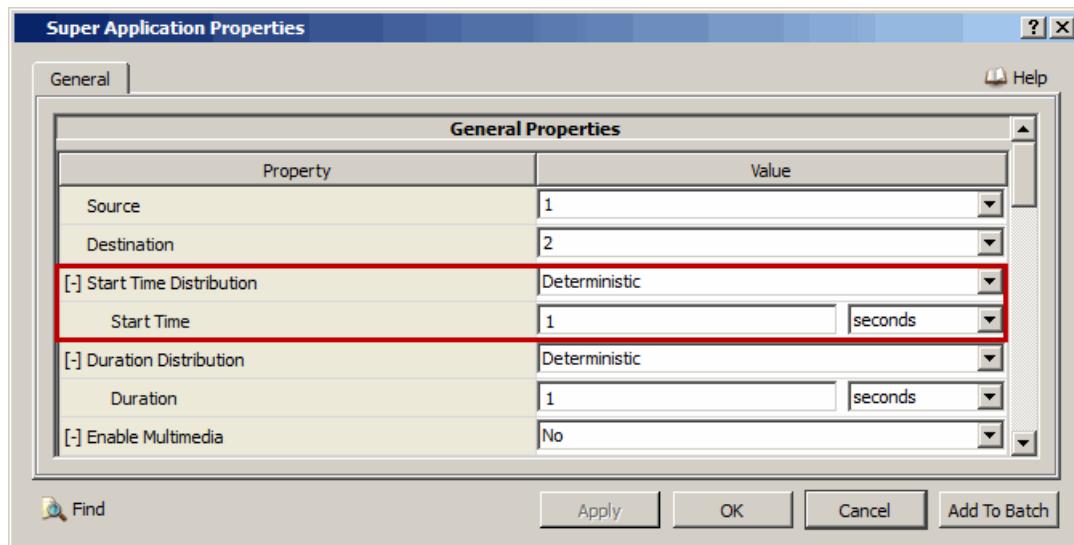
| GUI Parameter | Command Line Parameter |
|--|-------------------------|
| Source | <source> |
| Destination | <destination> |
| Start Time Distribution and dependent parameters | START-TIME <start-time> |

TABLE 8-42. Command Line Equivalent of Supper Application Parameters (Continued)

| GUI Parameter | Command Line Parameter |
|--|-------------------------------------|
| Duration Distribution and dependent parameters | DURATION <duration> |
| Enable Multimedia | N/A |
| Application Name | APPLICATION-NAME <application-name> |
| Chain ID | CHAIN-ID <chain-id> |
| Specify Conditions | CONDITIONS <conditions> |

Setting Parameters

- To specify an IP address as the source (destination) ID, set **Source (Destination)** to one of the IP addresses listed in the drop-down list.
 - To specify multimedia traffic (video or voice), set **Enable Multimedia** to Yes; otherwise, set **Enable Multimedia** to No.
5. To configure the start time parameters, set **Start Time Distribution** to *Deterministic*, *Exponential*, or *Uniform*.
- If **Start Time Distribution** is set to *Deterministic*, then set the dependent parameters listed in Table 8-43.

**FIGURE 8-17. Setting Parameters for a Deterministic Distribution****TABLE 8-43. Command Line Equivalent of Deterministic Distribution Parameters**

| GUI Parameter | Command Line Parameter |
|--|------------------------|
| Start Time Distribution (set to <i>Deterministic</i>) | DET |
| Start Time | <det-val> |

- If **Start Time Distribution** is set to *Exponential*, then set the dependent parameters listed in Table 8-44.

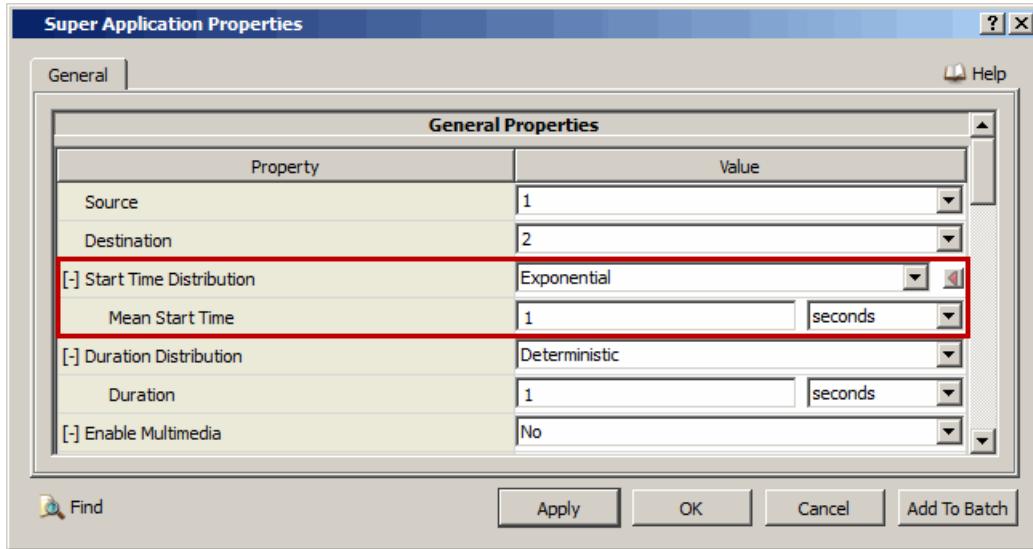


FIGURE 8-18. Setting Parameters for an Exponential Distribution

TABLE 8-44. Command Line Equivalent of Exponential Distribution Parameters

| GUI Parameter | Command Line Parameter |
|--|------------------------|
| Start Time Distribution (set to <i>Exponential</i>) | EXP |
| Mean Start Time | <exp-val> |

- If **Start Time Distribution** is set to *Uniform* then set the dependent parameters listed in [Table 8-45](#).

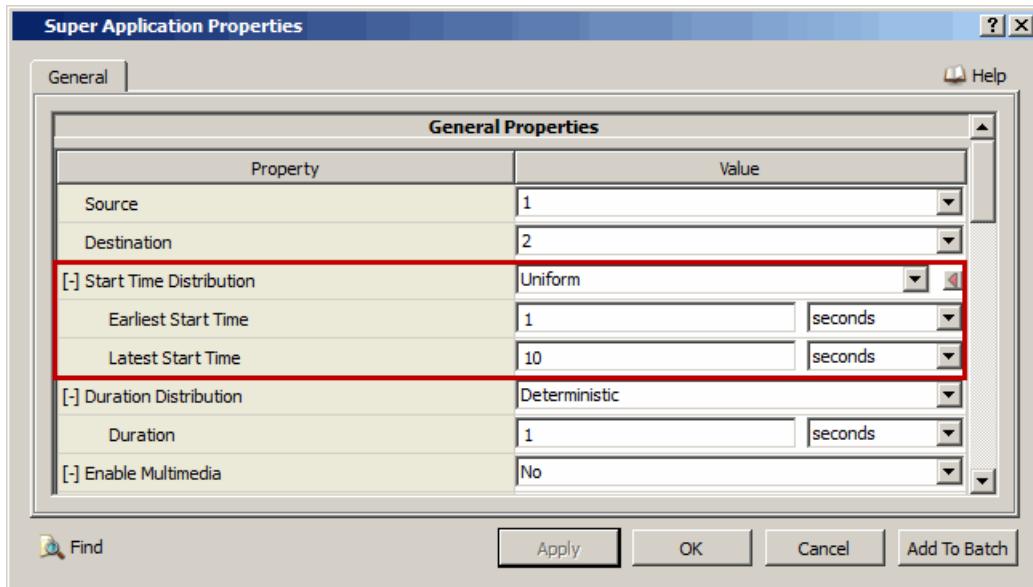


FIGURE 8-19. Setting Parameters for a Uniform Distribution

TABLE 8-45. Command Line Equivalent of Uniform Distribution Parameters

| GUI Parameter | Command Line Parameter |
|--|------------------------|
| Start Time Distribution (set to <i>Uniform</i>) | UNI |
| Earliest Start Time | <uni-val-1> |
| Latest Start Time | <uni-val-2> |

6. To configure the duration, set **Duration Distribution** and its dependent parameters in the same way as the start time parameters.

7. If **Enable Multimedia** is set to *No*, the set the data traffic parameters listed in [Table 8-46](#).

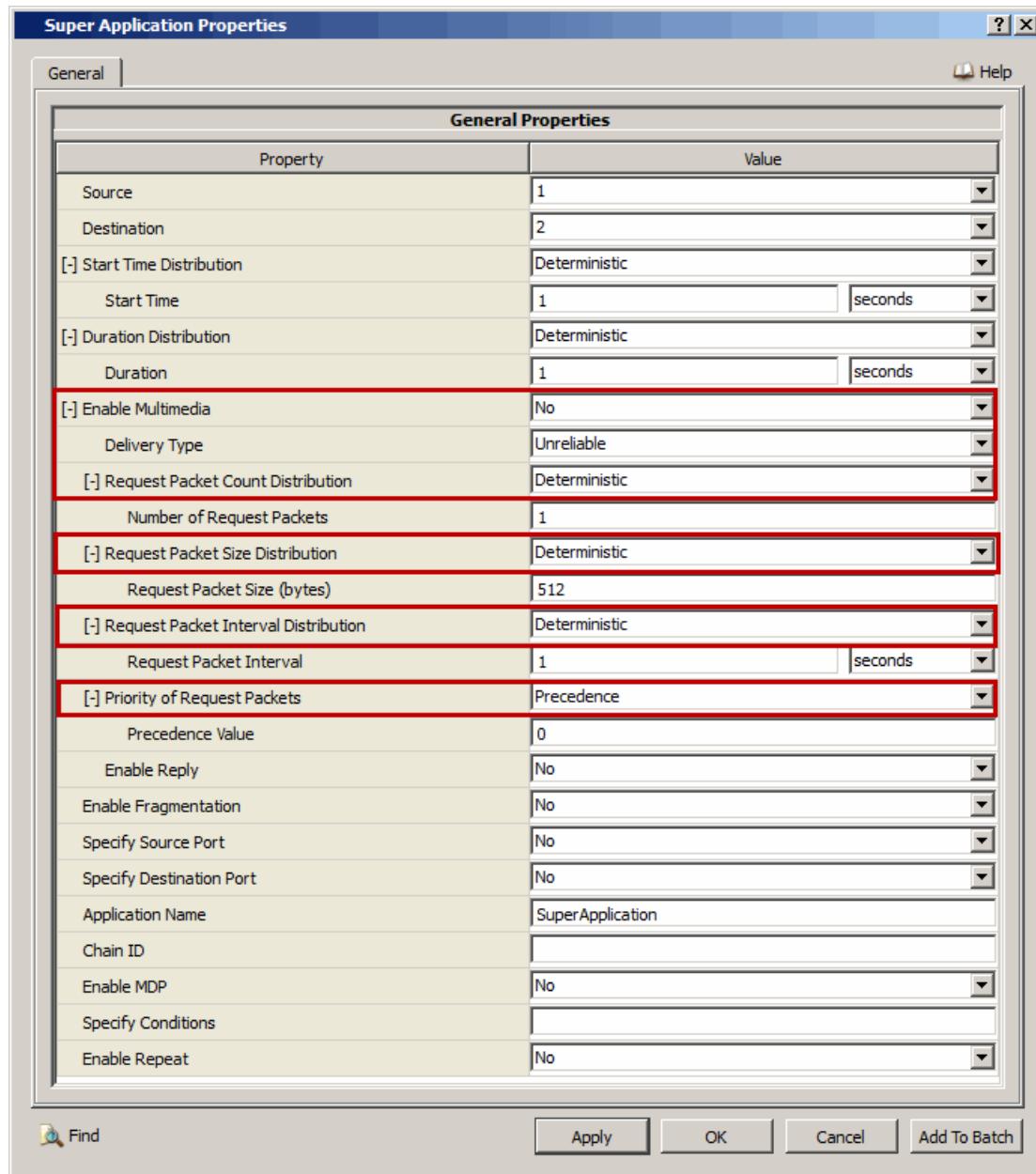


FIGURE 8-20. Setting Data Traffic Parameters

TABLE 8-46. Command Line Equivalent of Data Traffic Parameters

| GUI Parameter | Command Line Parameter |
|--|-------------------------------|
| Delivery Type | DELIVERY-TYPE <delivery-type> |
| Request Packet Count Distribution and dependent parameters | REQUEST-NUM <request-num> |

TABLE 8-46. Command Line Equivalent of Data Traffic Parameters (Continued)

| GUI Parameter | Command Line Parameter |
|---|-------------------------------------|
| Request Packet Size Distribution and dependent parameters | REQUEST-SIZE <request-size> |
| Request Packet Interval Distribution and dependent parameters | REQUEST-INTERVAL <request-interval> |
| Priority of Request Packets and dependent parameters | REQUEST-TOS <request-QoS> |

Setting Parameters

- In the GUI, the request data rate (corresponding to parameter REQUEST-DATA-RATE <data-rate>) can not be specified. The rate of the request traffic must be specified by using the parameters Request Packet Size Distribution and Request Packet Interval Distribution.
- To set DSCP, Precedence, or TOS bits for request packets, set **Priority of Request packets** to *DSCP*, *Precedence*, or *TOS*, respectively.
- To configure the number of request packets, set **Request Packet Count Distribution** and its dependent parameters in the same way as the start time parameters.
- To configure the size of request packets, set **Request Packet Size Distribution** and its dependent parameters in the same way as the start time parameters.
- To configure the time between request packets, set **Request Packet Interval Distribution** and its dependent parameters in the same way as the start time parameters.

8. If **Priority of Request Packets** is set to *DSCP*, *Precedence*, or *TOS*, set the dependent parameters listed in [Table 8-47](#), [Table 8-48](#), and [Table 8-49](#), respectively. [Figure 8-21](#) shows how to set the dependent parameters when **Priority of Request Packets** is set to *DSCP*. Setting dependent parameters for the other two options is similar.

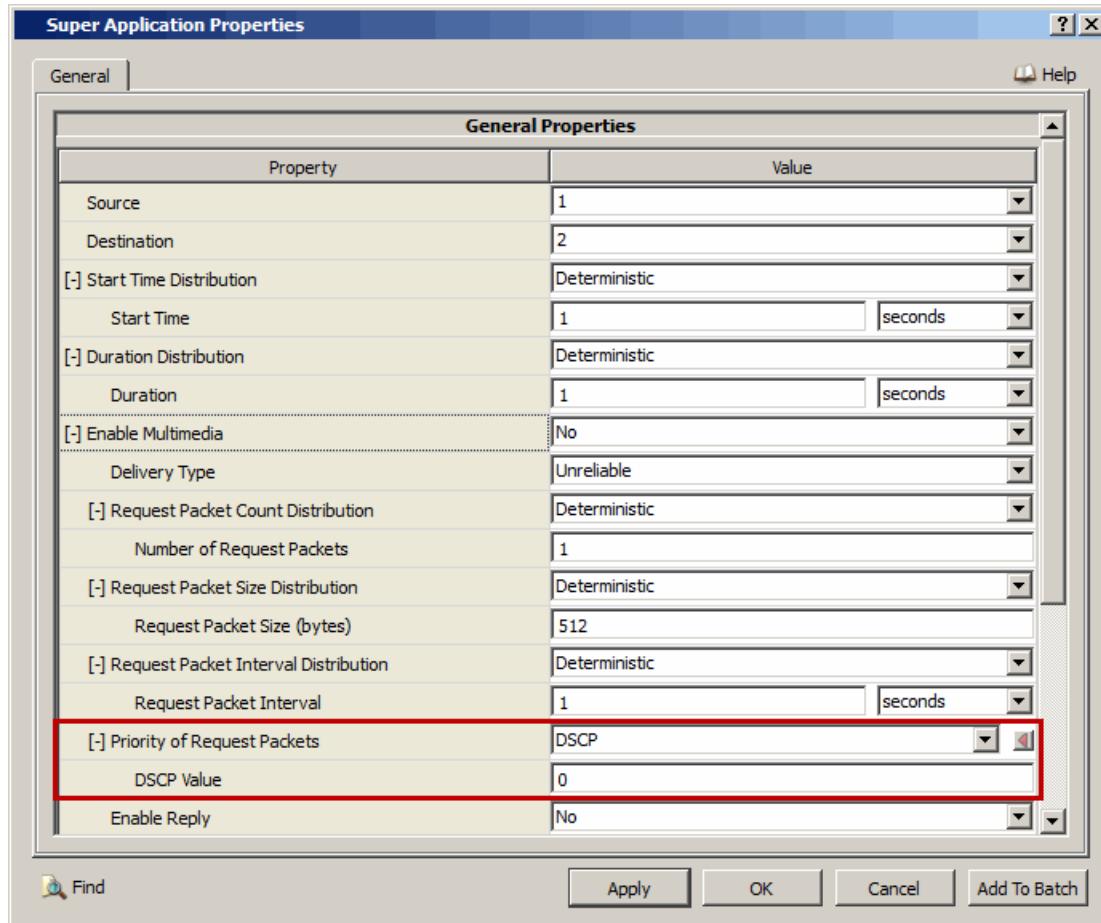


FIGURE 8-21. Setting DSCP Value

TABLE 8-47. Command Line Equivalent of Priority Parameters (Priority of Request Packets = DSCP)

| GUI Parameter | Command Line Parameter |
|------------------------|------------------------|
| Priority (set to DSCP) | DSCP |
| DSCP Value | <DSCP-value> |

TABLE 8-48. Command Line Equivalent of Priority Parameters (Priority of Request Packets = Precedence)

| GUI Parameter | Command Line Parameter |
|--------------------------------------|------------------------|
| Priority (set to <i>Precedence</i>) | PRECEDENCE |
| Precedence Value | <precedence-value> |

TABLE 8-49. Command Line Equivalent of Priority Parameters (Priority of Request Packets = TOS)

| GUI Parameter | Command Line Parameter |
|-------------------------------|------------------------|
| Priority (set to <i>TOS</i>) | TOS |
| TOS Value | <TOS-value> |

9. To specify the reply packet parameters, set **Enable Reply** to Yes and set the dependent parameters listed in Table 8-50.

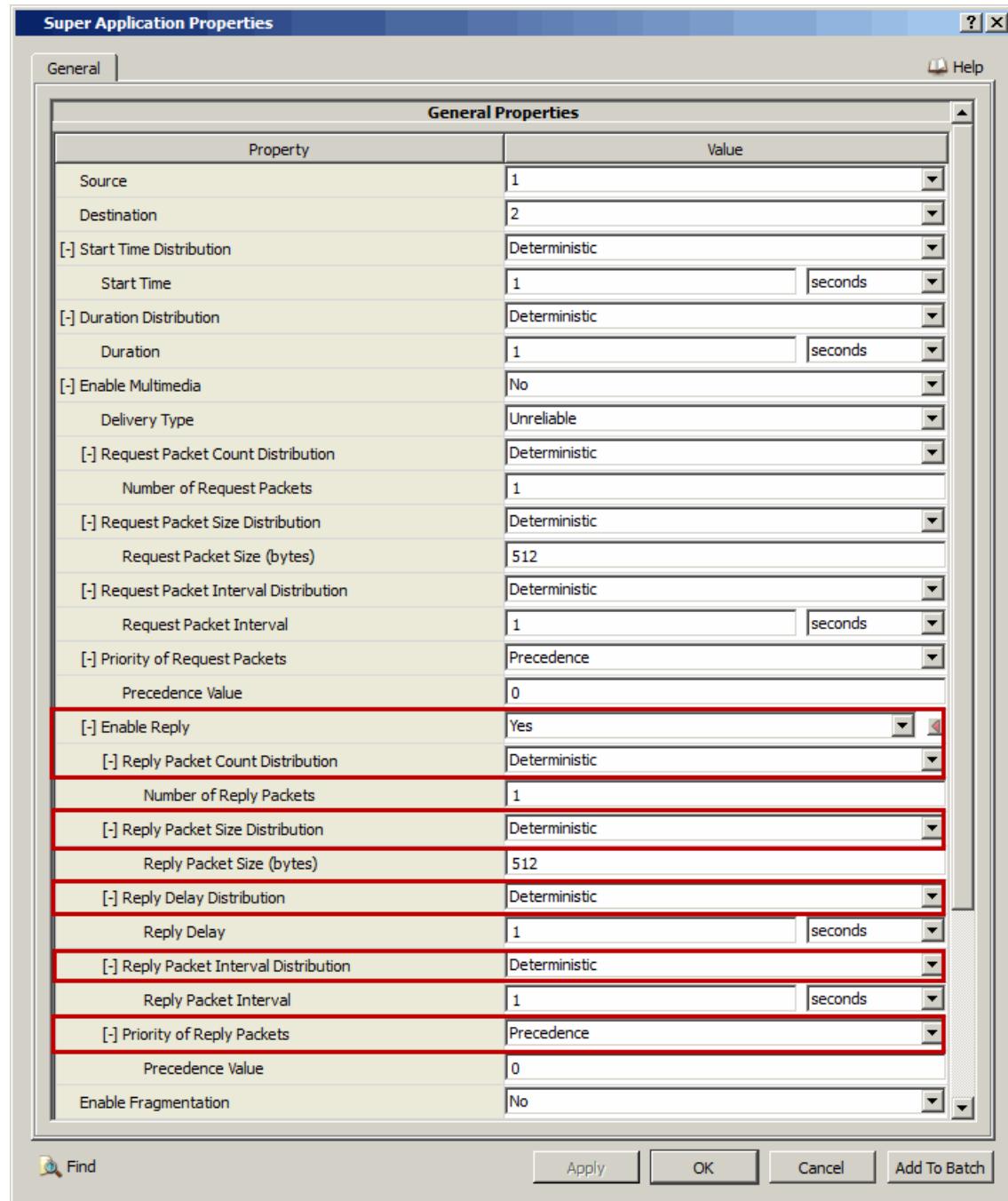


FIGURE 8-22. Setting Reply Packet Parameters

TABLE 8-50. Command Line Equivalent of Reply Packet Parameters

| GUI Parameter | Command Line Parameter |
|---|--|
| Enable Reply (set to Yes) | REPLY-PROCESS YES |
| Enable Reply (set to No) | REPLY-PROCESS NO |
| Reply Packet Count Distribution and dependent parameters | REPLY-NUM <reply-num> |
| Reply Packet Size Distribution and dependent parameters | REPLY-SIZE <reply-size> |
| Reply Delay Distribution and dependent parameters | REPLY-PROCESS-DELAY <reply-delay> |
| Reply Packet Interval Distribution and dependent parameters | REPLY-INTERDEPARTURE-DELAY <reply-delay> |
| Priority of Reply Packets and dependent parameters | REPLY-TOS <reply-QoS> |

Setting Parameters

- To configure the number of reply packets, set **Reply Packet Count Distribution** and its dependent parameters in the same way as the start time parameters.
- To configure the size of reply packets, set **Reply Packet Size Distribution** and its dependent parameters in the same way as the start time parameters.
- To configure the delay for reply packets, set **Reply Delay Distribution** and its dependent parameters in the same way as the start time parameters.
- To configure the time between reply packets, set **Reply Packet Interval Distribution** and its dependent parameters in the same way as the start time parameters.
- Set **Priority of Reply Packets** and its dependent parameters in the same way as **Priority of Request Packets** and its dependent parameters.

10. If **Enable Multimedia** is set to Yes, the set the multimedia traffic parameters listed in [Table 8-51](#).

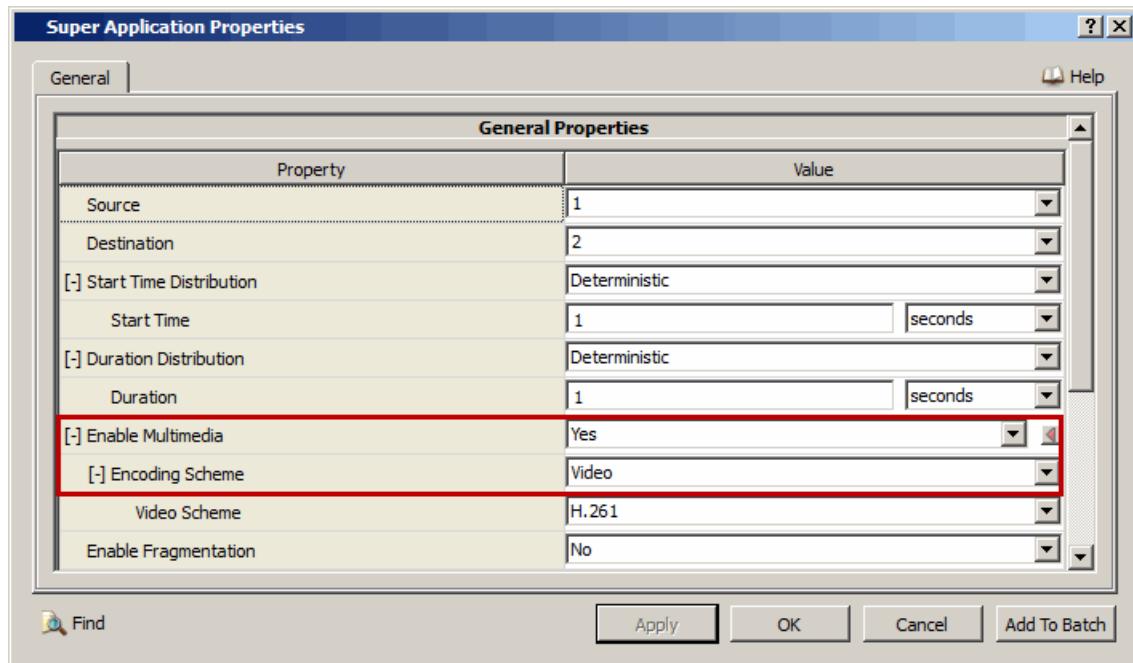


FIGURE 8-23. Enabling Multimedia Traffic

TABLE 8-51. Command Line Equivalent of Multimedia Traffic Parameters

| GUI Parameter | Command Line Parameter |
|--------------------------------|------------------------|
| Enable Multimedia (set to Yes) | ENCODING-SCHEME |
| Encoding Scheme | <traffic-type> |

11. If **Encoding Scheme** is set to *Video*, the set the dependent parameters listed in [Table 8-52](#).

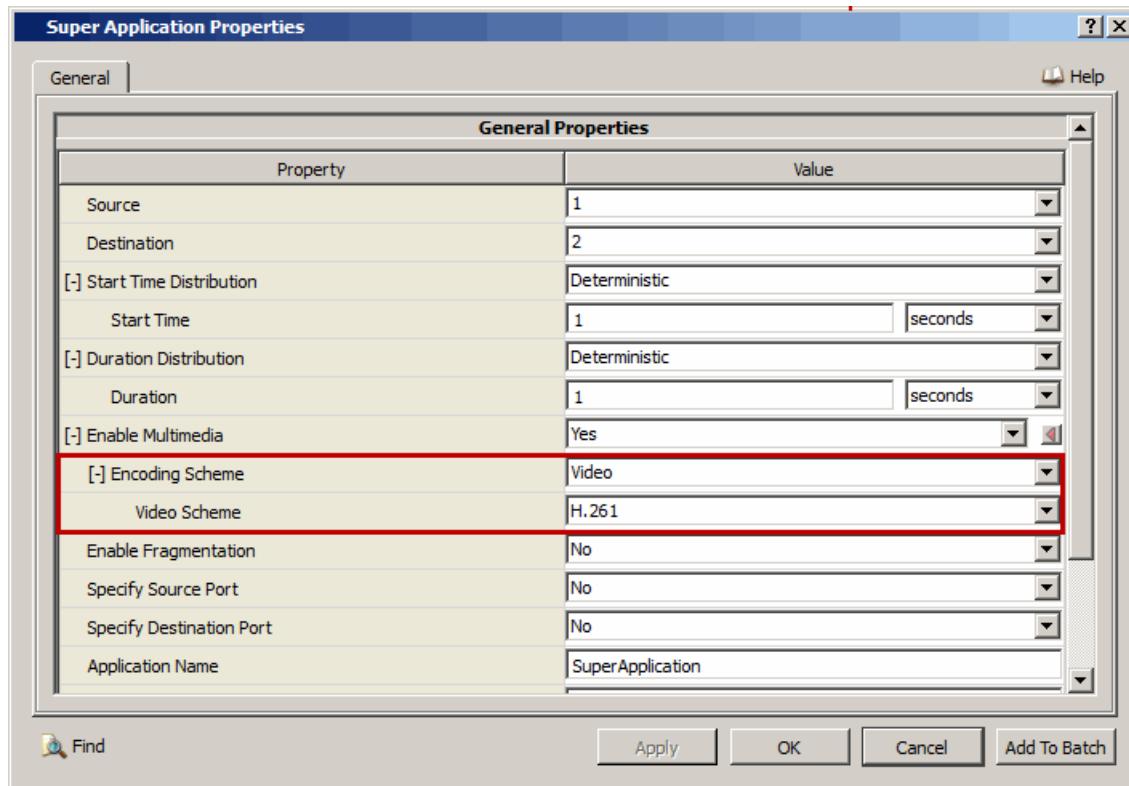


FIGURE 8-24. Enabling Video Traffic

TABLE 8-52. Command Line Equivalent of Video Traffic Parameters

| GUI Parameter | Command Line Parameter |
|---------------|------------------------|
| Video Scheme | <codec> |

12. If **Encoding Scheme** is set to **Voice**, the set the dependent parameters listed in [Table 8-53](#).

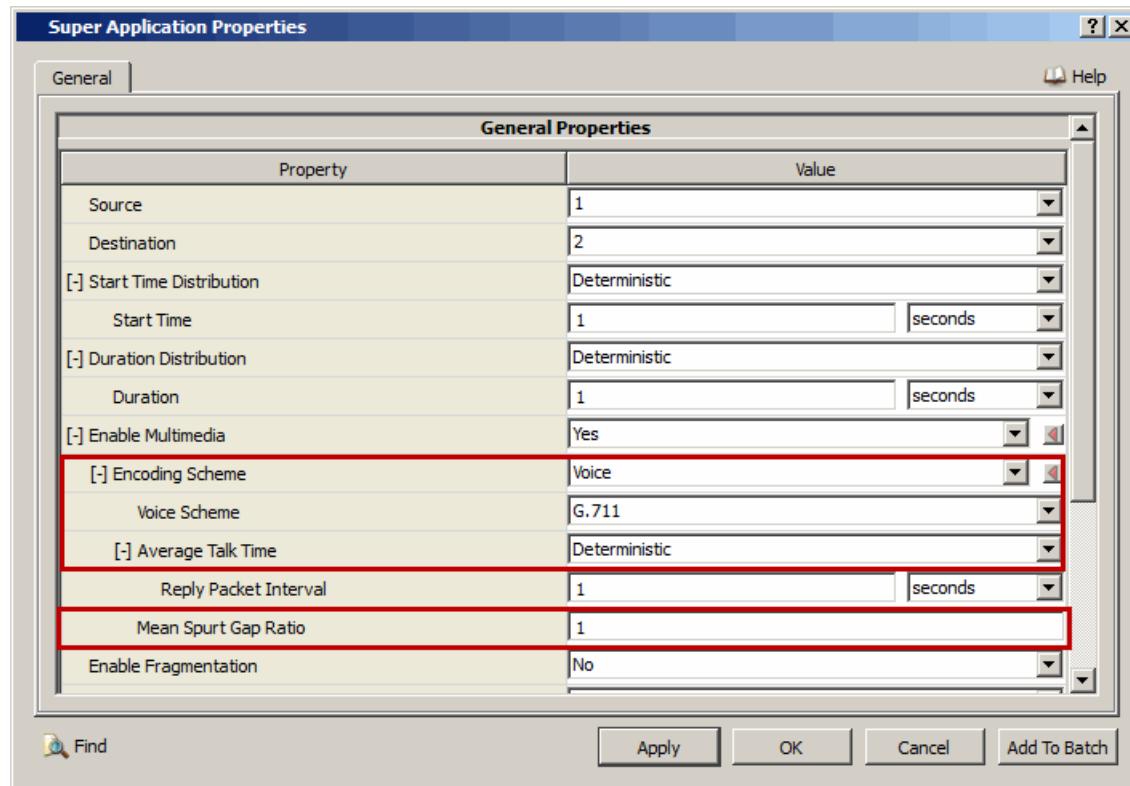


FIGURE 8-25. Enabling Voice Traffic

TABLE 8-53. Command Line Equivalent of Voice Traffic Parameters

| GUI Parameter | Command Line Parameter |
|--|-----------------------------------|
| Voice Scheme | <codec> |
| Average Talk Time and dependent parameters | AVERAGE-TALK-TIME <avg-talk-time> |
| Mean Spurt Gap Ratio | MEAN-SPURT-GAP-RATIO <gap-ratio> |

- To configure the average talk time, set **Average Talk Time** and its dependent parameters in the same way as the start time parameters.

13. To specify the fragment size, set **Enable Fragmentation** Yes and set the dependent parameters listed in Table 8-54.

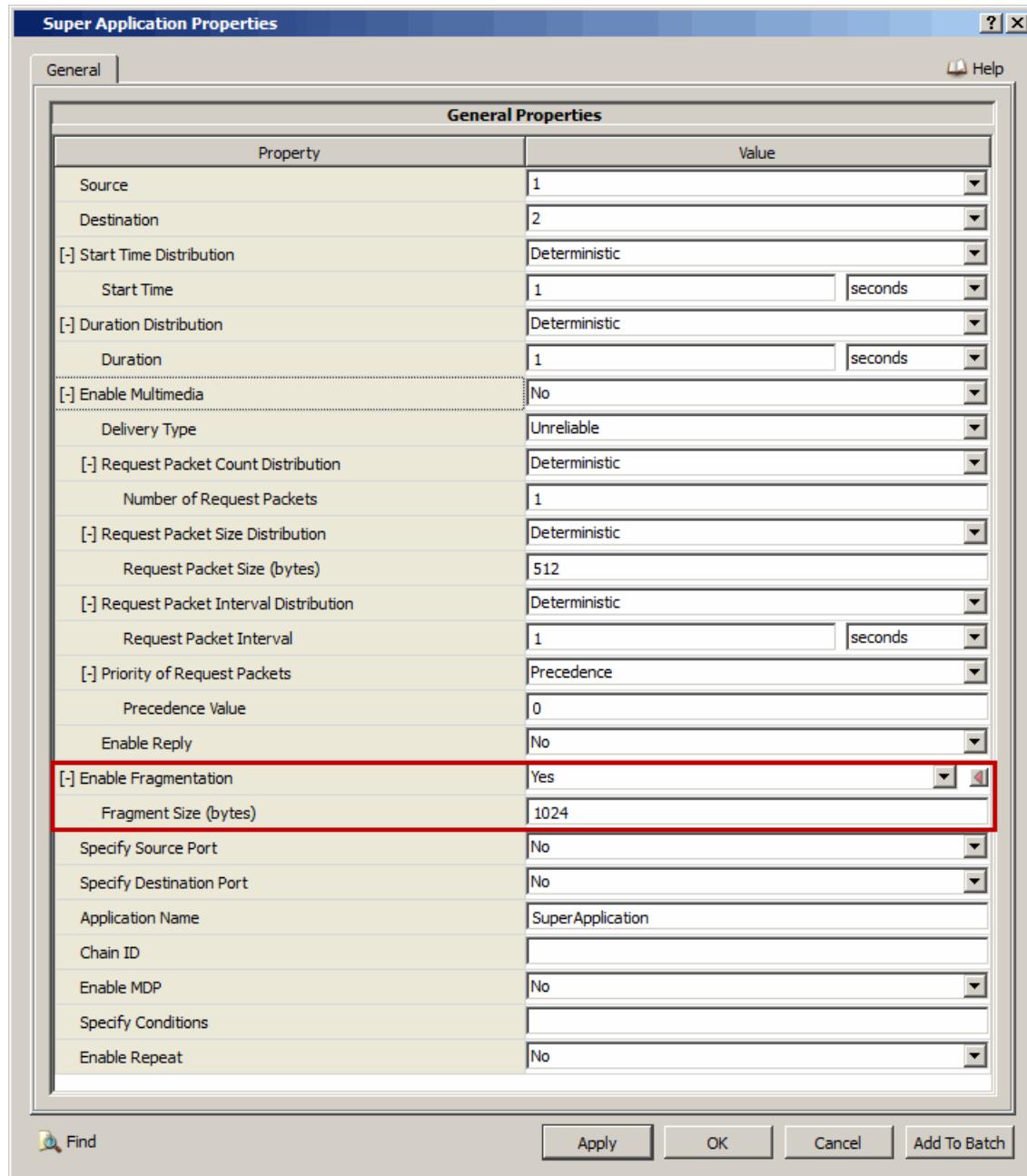


FIGURE 8-26. Setting Fragmentation Parameters

TABLE 8-54. Command Line Equivalent of Fragmentation Parameters

| GUI Parameter | Command Line Parameter |
|-----------------------------------|------------------------|
| Enable Fragmentation (set to Yes) | FRAGMENT-SIZE |
| Fragment Size | <fragment-size> |

14. To specify the source port, set **Specify Source Port** to Yes and set the dependent parameters listed in Table 8-55.

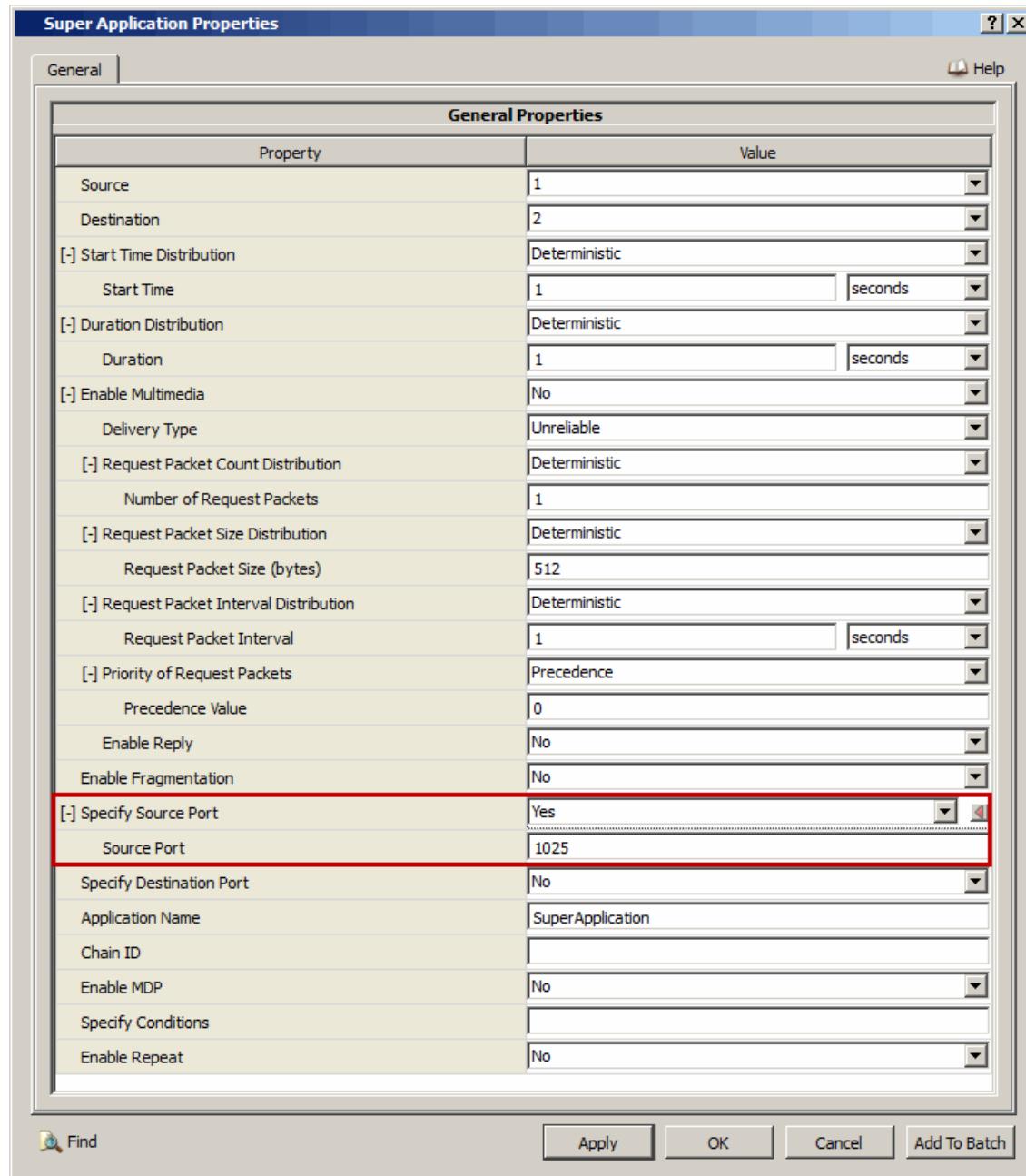


FIGURE 8-27. Setting Source Port Parameters

TABLE 8-55. Command Line Equivalent of Source Port Parameters

| GUI Parameter | Command Line Parameter |
|----------------------------------|------------------------|
| Specify Source Port (set to Yes) | SOURCE-PORT |
| Source Port | <source-port> |

15. To specify the destination port, set **Specify Destination Port** to Yes and set the dependent parameters listed in Table 8-56.

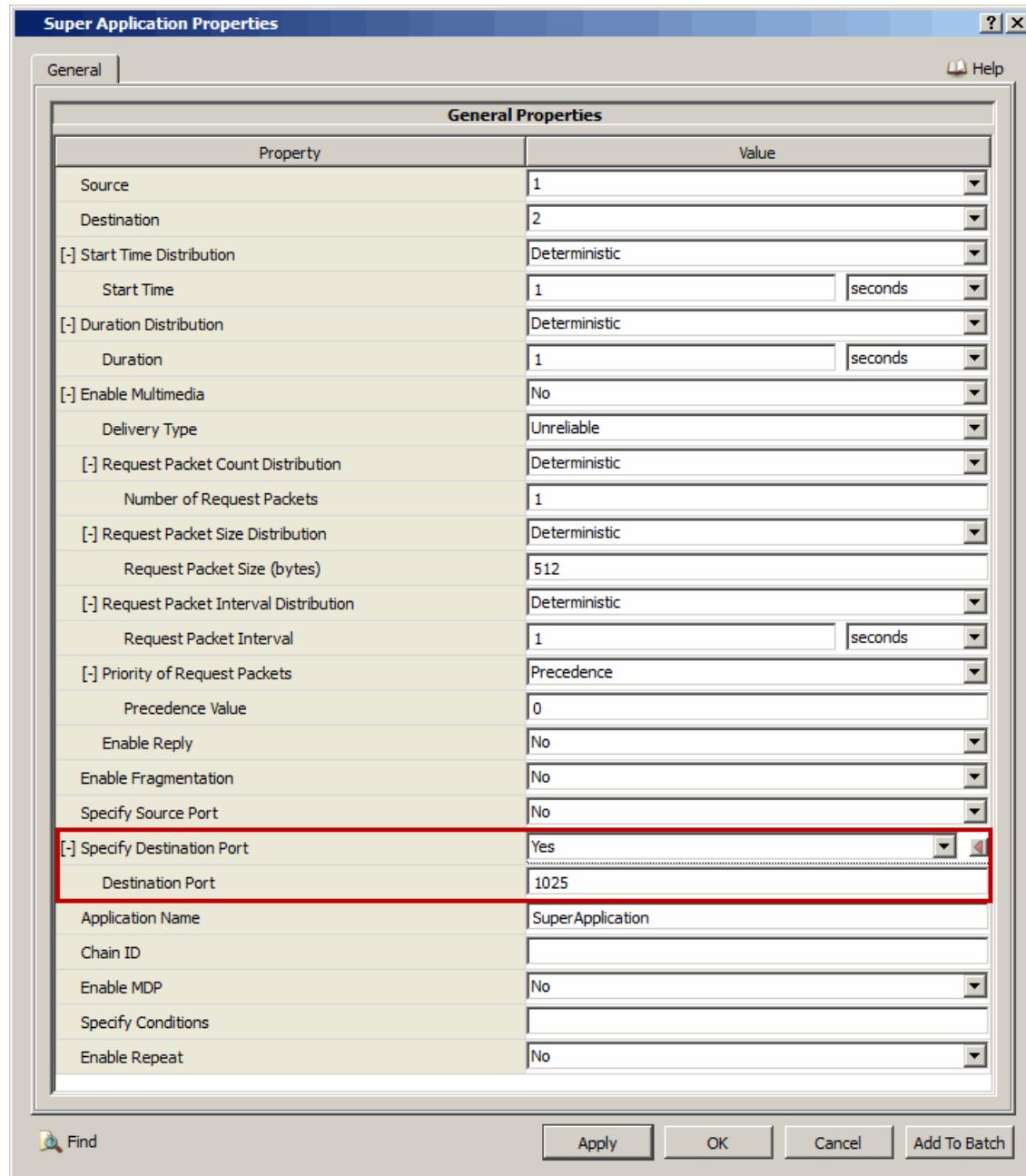


FIGURE 8-28. Setting Destination Port Parameters

TABLE 8-56. Command Line Equivalent of Destination Port Parameters

| GUI Parameter | Command Line Parameter |
|---------------------------------------|------------------------|
| Specify Destination Port (set to Yes) | DESTINATION-PORT |
| Destination Port | <destination-port> |

16. To configure the MDP parameters, perform the following steps:

- To enable MDP, set **Enable MDP** to Yes; otherwise, set **Enable MDP** to No.

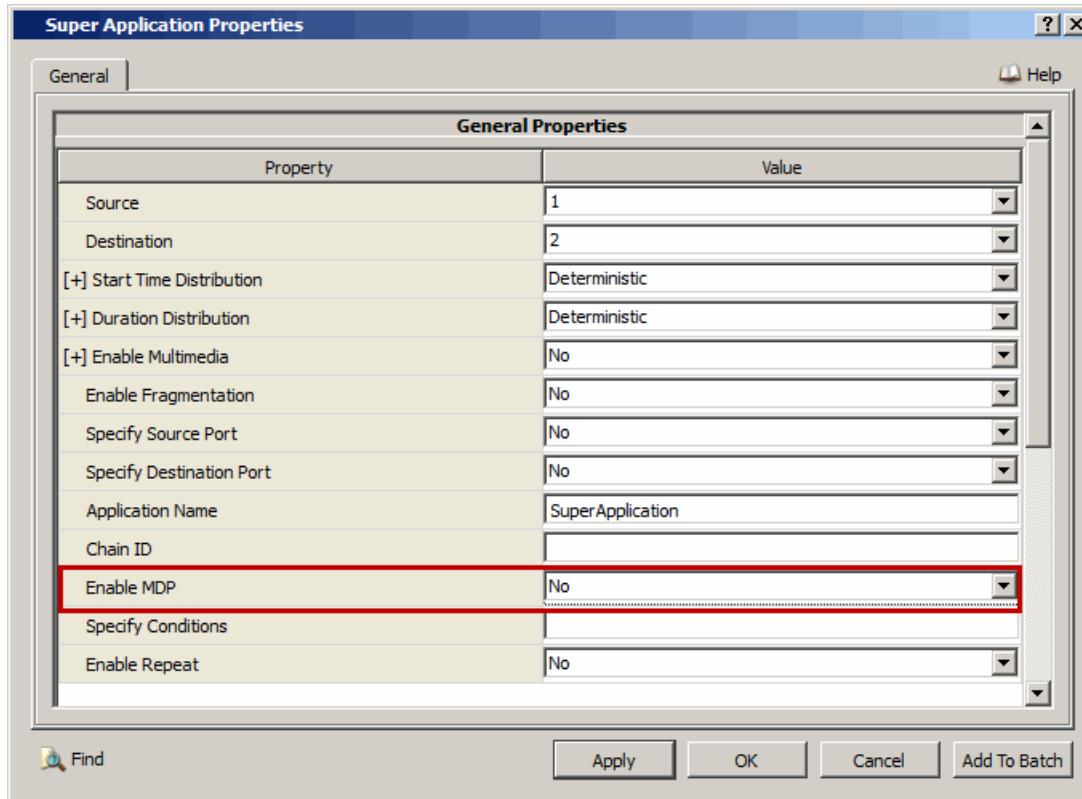


FIGURE 8-29. Enabling MDP

TABLE 8-57. Command Line Equivalent of MDP Parameters

| GUI Parameter | Command Line Parameter |
|-------------------------|------------------------|
| Enable MDP (set to Yes) | MDP - ENABLED |

- b. If **Enable MDP** is set to Yes, then set the dependent parameters listed in Table 8-58.

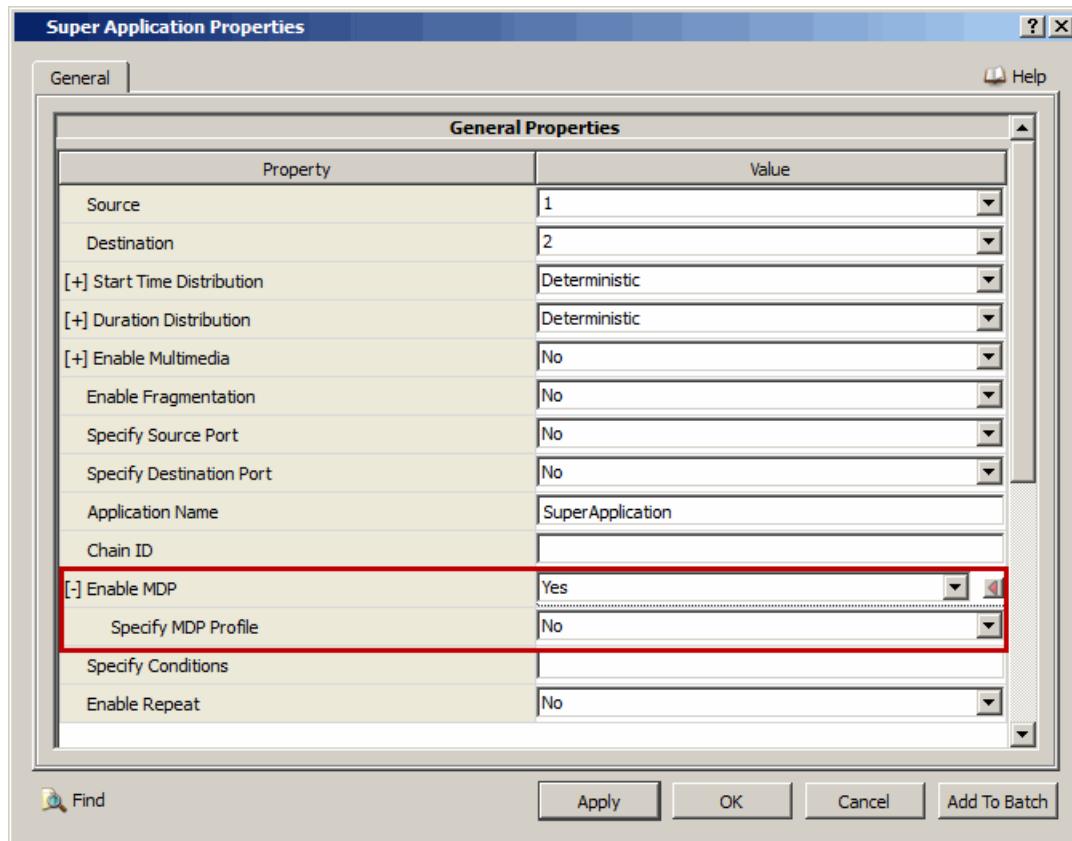


FIGURE 8-30. Configuring MDP Parameters

TABLE 8-58. Command Line Equivalent of MDP Configuration Parameters

| GUI Parameter | Command Line Parameter |
|----------------------------------|------------------------|
| Specify MDP Profile (set to Yes) | MDP - PROFILE |

Setting Parameters

- To specify an MDP profile, set **Specify MDP Profile** to Yes. To use the default MDP profile, set **Specify MDP Profile** to No.

- c. If **Specify MDP Profile** is set to Yes, then set the parameters listed in [Table 8-59](#).

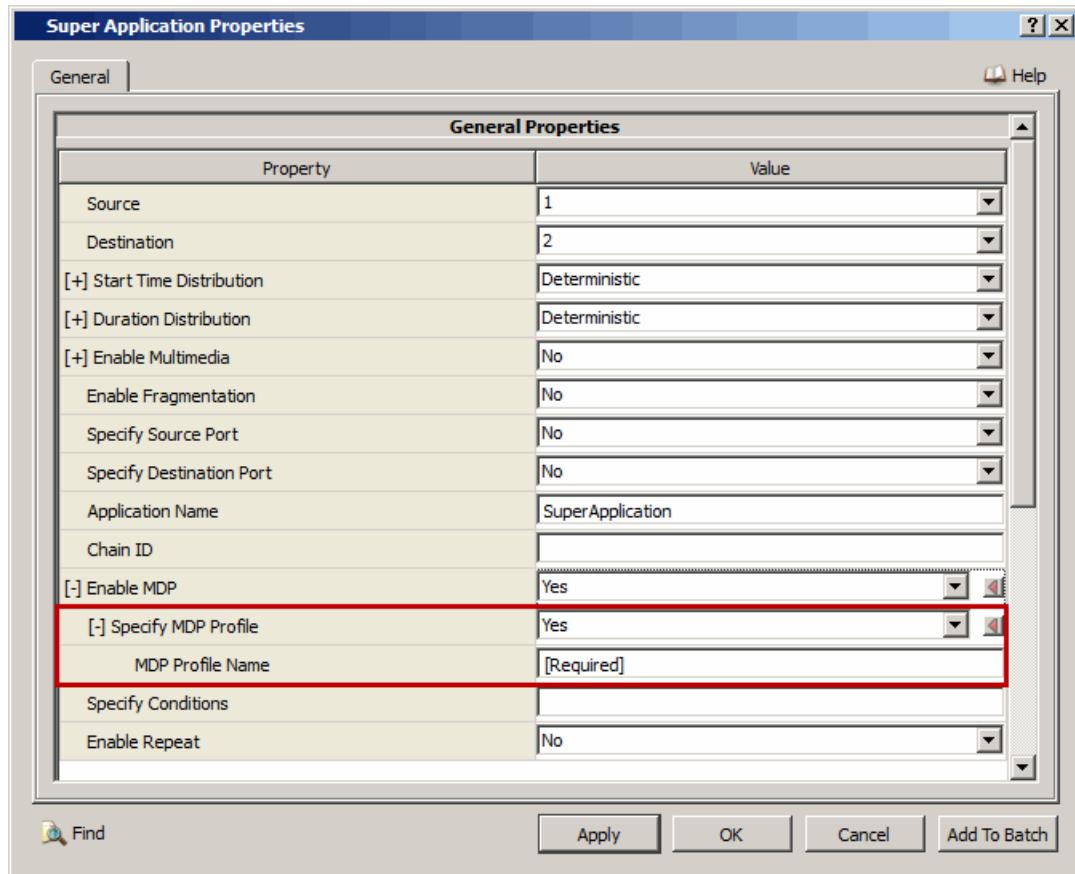


FIGURE 8-31. Specifying MDP Profile

TABLE 8-59. Command Line Equivalent of MDP Profile Parameters

| GUI Parameter | Command Line Parameter |
|------------------|------------------------|
| MDP Profile Name | <profile-name> |

17. To enable repetition of the application flow, set **Enable Repeat** to Yes and set the dependent parameters listed in [Table 8-60](#).

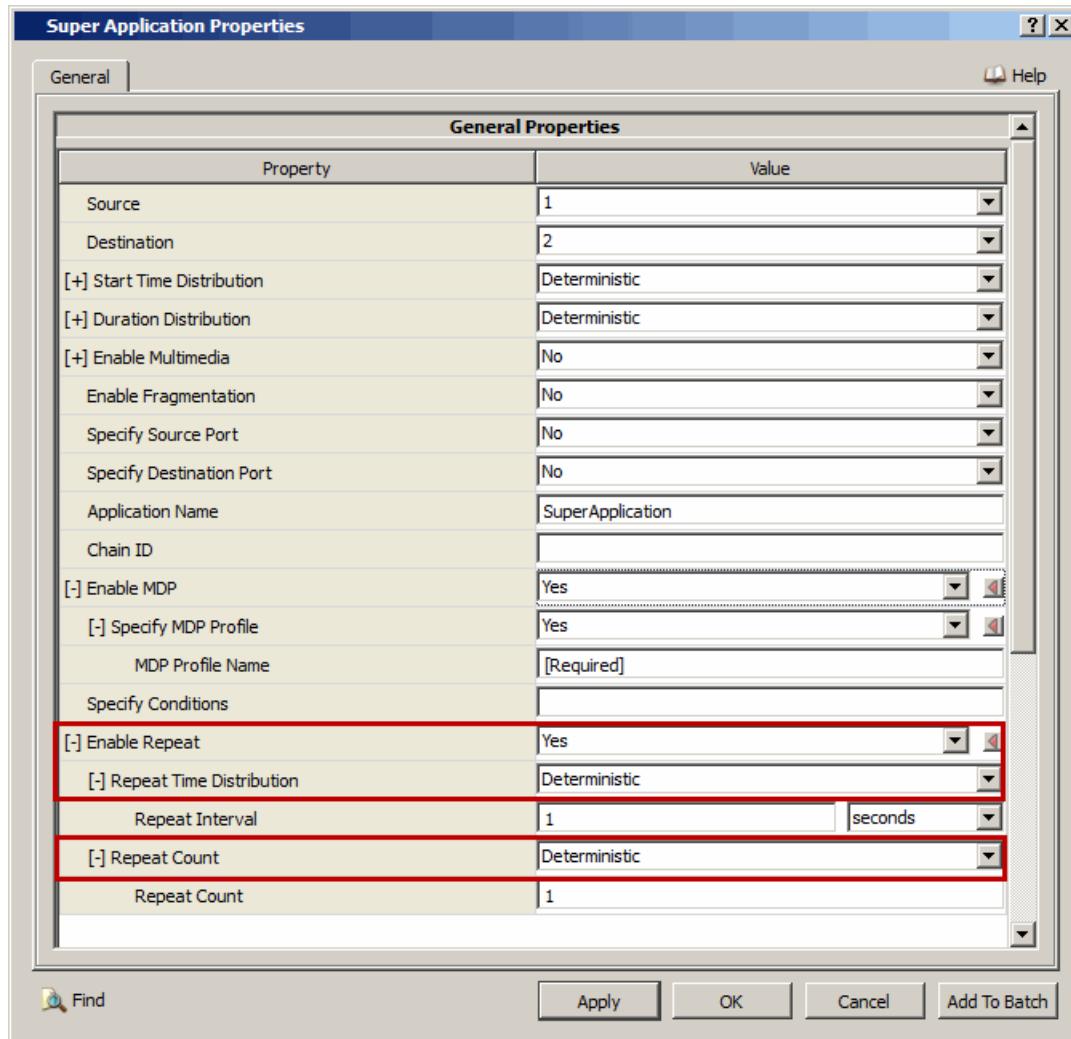


FIGURE 8-32. Enabling Application Flow Repetitions

TABLE 8-60. Command Line Equivalent of Application Flow Repetition Parameters

| GUI Parameter | Command Line Parameter |
|---|------------------------|
| Enable Repeat (set to Yes) | REPEAT |
| Repeat Time Distribution and dependent parameters | <repeat-interval> |
| Repeat Count and dependent parameters | <no-of-repeats> |

8.7.4.2 Configuring a Single Host Super Application Session

To configure a single host Super Application session, perform the following steps:

1. Click the **SUPR APP** button in the **Single Host Applications** tab of the Standard Toolset.
2. On the canvas, click on the client node.
3. Open the Super Application Properties Editor by doing one of the following:
 - On the canvas, either double-click on the  symbol next to the client node or right-click on the  symbol and select **Properties** from the menu.
 - In the **Applications** tab of Table View, either double-click on the application row or right-click on the application row and select **Properties** from the menu.
4. Set the source and destination parameters listed in [Table 8-61](#).

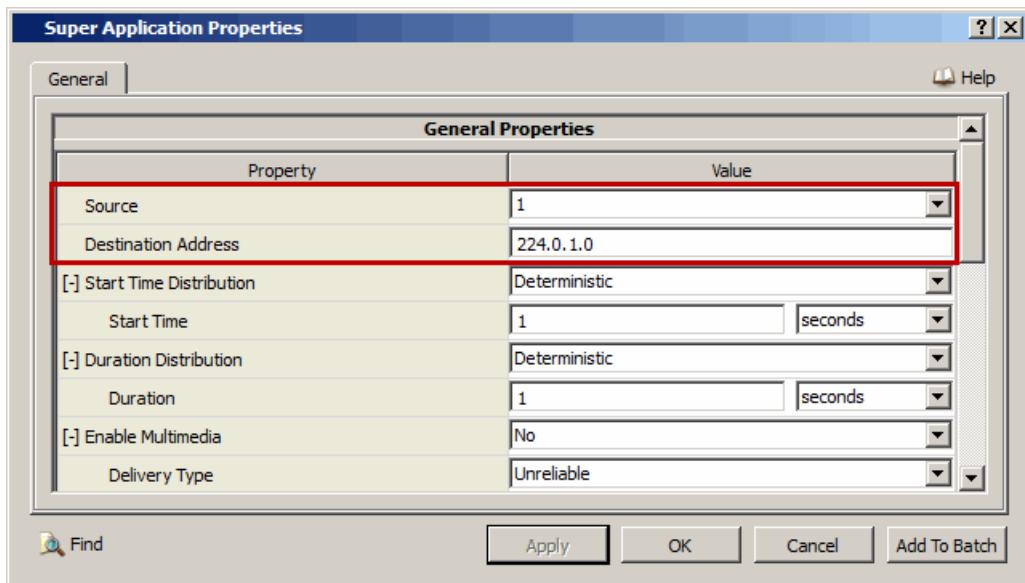


FIGURE 8-33. Setting Single Host Source and Destination Parameters

TABLE 8-61. Command Line Equivalent of Single Host Source and Destination Parameters

| GUI Parameter | Command Line Parameter |
|-------------------------|------------------------|
| Source | <Source> |
| Multicast Group Address | <Destination> |

Setting Parameters

- To specify an IP address as the source ID, set **Source** to one of the IP addresses listed in the drop-down list.
 - Set **Destination Address** to the address of the multicast group that is to receive traffic from the source.
5. Set the remaining application parameters as for a client-server session (see [Section 8.7.4.1](#)).

8.7.4.3 Configuring Statistics Parameters

Statistics for applications (including Super Application) can be collected at the global and node levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for Super Application, check the box labeled **Application** in the appropriate properties editor.

TABLE 8-62. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| Application | Global, Node | APPLICATION-STATISTICS |

8.7.5 Statistics

This section describes the file, database, and dynamic statistics of the Super Application model.

8.7.5.1 File Statistics

Table 8-63 shows the Super Application statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 8-63. Super Application Statistics

| Statistic | Description |
|---|--|
| Super Application Server | |
| Server Address | The address of the Server. |
| Connection: Client(Source addr: Source port) - Server(Dest addr: Dest port) | Connection ID (Node numbers and ports). |
| Multicast Session Start (seconds) | Time in second when multicast session was started. |
| Multicast Session Finish (seconds) | Time in second when multicast session was finished. |
| First Multicast Fragment Sent (seconds) | Time in second when first multicast fragment was sent. |
| Last Multicast Fragment Sent (seconds) | Time in second when last multicast fragment was sent. |
| First Multicast Fragment Received (seconds) | Time in second when first multicast fragment was received. |
| Last Multicast Fragment Received (seconds) | Time in second when last multicast fragment was received |
| Total Multicast Fragments Sent (fragments) | Total number of multicast fragments sent. |
| Total Multicast Fragments Received (fragments) | Total number of multicast fragments received. |
| First Multicast Message Sent (seconds) | Time in second when first multicast message was sent. |
| Last Multicast Message Sent (seconds) | Time in second when last multicast message was sent. |
| First Multicast Message Received (seconds) | Time in second when first multicast message was received. |
| Last Multicast Message Received (seconds) | Time in second when last multicast message was received. |
| Total Multicast Messages Sent (messages) | Total number of multicast messages sent. |
| Total Multicast Messages Received (messages) | Total number of multicast messages received. |
| Total Multicast Data Sent (bytes) | Total number of multicast data bytes sent. |
| Total Multicast Data Received (bytes) | Total number of multicast data bytes received. |
| Total Multicast Overhead Sent (bytes) | Total number of multicast overhead bytes sent. |
| Total Multicast Overhead Received (bytes) | Total number of multicast overhead bytes received. |
| Average Multicast End-to-End Delay (seconds) | Average multicast end-to-end delay. |

TABLE 8-63. Super Application Statistics (Continued)

| Statistic | Description |
|---|--|
| Multicast Offered Load (bits/second) | Multicast offered load. |
| Multicast Received Throughput (bits/second) | Multicast received throughput. |
| Smoothed Multicast Jitter (seconds) | Smoothed multicast jitter. |
| Average Multicast Jitter (seconds) | Average multicast jitter. |
| Total Multicast Jitter (seconds) | Total multicast jitter. |
| Number of out of order reply fragments received | Total number of out of order reply fragments received. |
| Number of complete reply packets received | Total number of complete reply packets received. |
| Number of out of order request fragments received | Total number of out of order request fragments received. |
| Number of complete request packets received | Total number of complete request packets received. |
| Super Application Client | |
| Client Address | The address of the client. |
| Connection: Client(Source addr: Source port) - Server(Dest addr: Dest port) | Connection ID (Node numbers and ports) |
| Multicast Session Start (seconds) | Time in second when multicast session was started. |
| Multicast Session Finish (seconds) | Time in second when multicast session was finished. |
| First Multicast Fragment Sent (seconds) | Time in second when first multicast fragment was sent. |
| Last Multicast Fragment Sent (seconds) | Time in second when last multicast fragment was sent. |
| First Multicast Fragment Received (seconds) | Time in second when first multicast fragment was received. |
| Last Multicast Fragment Received (seconds) | Time in second when last multicast fragment was received |
| Total Multicast Fragments Sent (fragments) | Total number of multicast fragments sent. |
| Total Multicast Fragments Received (fragments) | Total number of multicast fragments received. |
| First Multicast Message Sent (seconds) | Time in second when first multicast message was sent. |
| Last Multicast Message Sent (seconds) | Time in second when last multicast message was sent. |
| First Multicast Message Received (seconds) | Time in second when first multicast message was received. |
| Last Multicast Message Received (seconds) | Time in second when last multicast message was received. |
| Total Multicast Messages Sent (messages) | Total number of multicast messages sent. |
| Total Multicast Messages Received (messages) | Total number of multicast messages received. |
| Total Multicast Data Sent (bytes) | Total number of multicast data bytes sent. |
| Total Multicast Data Received (bytes) | Total number of multicast data bytes received. |
| Total Multicast Overhead Sent (bytes) | Total number of multicast overhead bytes sent. |
| Total Multicast Overhead Received (bytes) | Total number of multicast overhead bytes received. |
| Average Multicast End-to-End Delay (seconds) | Average multicast end-to-end delay. |
| Multicast Offered Load (bits/second) | Multicast offered load. |
| Multicast Received Throughput (bits/second) | Multicast received throughput. |
| Smoothed Multicast Jitter (seconds) | Smoothed multicast jitter. |
| Average Multicast Jitter (seconds) | Average multicast jitter. |
| Total Multicast Jitter (seconds) | Total multicast jitter. |
| Number of out of order reply fragments received | Total number of out of order reply fragments received. |
| Number of complete reply packets received | Total number of complete reply packets received. |

TABLE 8-63. Super Application Statistics (Continued)

| Statistic | Description |
|---|--|
| Number of out of order request fragments received | Total number of out of order request fragments received. |
| Number of complete request packets received | Total number of complete request packets received. |

Notes: 1. The throughput for requests sent (bits/s) is calculated as follows:

$$\text{Throughput for requests sent (bits/s)} = (\text{Total number of request bits sent}) / (\text{Last request fragment sent time} - \text{First request fragment sent time})$$

2. The throughput for requests received (bits/s) is calculated as follows:

$$\text{Throughput for requests received (bits/s)} = (\text{Total number of request fragment bits received}) / (\text{Last request fragment receive time} - \text{First request fragment receive time})$$

3. The throughput for replies sent (bits/s) is calculated as follows:

$$\text{Throughput for replies sent (bits/s)} = (\text{Total number of reply bits sent}) / (\text{Last reply fragment sent time} - \text{First reply fragment sent time})$$

4. The throughput for replies received (bits/s) is calculated as follows:

$$\text{Throughput for replies received (bits/s)} = (\text{Total number of reply fragment bits received}) / (\text{Last reply fragment receive time} - \text{First reply fragment receive time})$$

8.7.5.2 Database Statistics

In addition to the file statistics, the Super Application model also enters statistics in various scenario statistics database tables. Refer to *QualNet Statistics Database User's Guide* for details.

8.7.5.3 Dynamic Statistics

The following dynamic statistics are enabled for the Super Application model (refer to Chapter 5 of *QualNet User's Guide* for details of viewing dynamic statistics in the GUI during the simulation):

- Number of Request Bytes Sent
- Number of Request Packets Sent
- Number of Request Fragments Sent
- Number of Reply Bytes Sent
- Number of Reply Packets Sent
- Number of Reply Fragments Sent
- Number of Out-of-order Reply Fragments Received
- Number of Complete Reply Packets Received
- Number of Reply Fragments Received
- Total Reply Fragment Bytes Received
- Number of Out-of-order Request Fragments Received
- Number of Complete Request Packets Received
- Number of Request Fragments Received
- Total Request Fragment Bytes Received
- Throughput for Requests Sent (bits/s)

- Throughput for Requests Received (bits/s)
- Throughput for Replies Sent (bits/s)
- Throughput for Replies Received (bits/s)
- Total Packet Real Time/Simulation Time Ratio
- Average Packet Real Time/Simulation Time Ratio
- Total Packets with Real Time Greater than Simulation Time

8.8 Telecommunications Network (TELNET)

8.8.1 Description

TELNET represents the clear-text terminal server and client. The typing rate and sizes of server responses are taken from distributions created from network traces in the TCPlib library.

8.8.2 Command Line Configuration

Application Configuration File Parameters

To specify TELNET traffic, include the following statement in the application configuration file (.app):

```
TELNET <src> <dest> <session-duration> <start-time>
```

Table 8-64 shows the TELNET parameters. See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 8-64. TELNET Parameters

| Parameter | Value | Description |
|--------------------------------|--|---|
| <src> Required | Integer or IP Address | Client node's ID or IP address. |
| <dest> Required | Integer or IP Address | Server node's ID or IP address. |
| <session-duration> Required | Time <i>Range:</i> $\geq 0\text{S}$ | Length of the entire session. |
| <start-time> Required | Time <i>Range:</i> $\geq 0\text{S}$ | Time when the transmission of packets should begin. |

Scenario Configuration File Parameters

[Table 8-65](#) describes the TELNET parameters that can be specified in the scenario configuration (.config) file.

TABLE 8-65. TELNET Scenario Configuration File Parameters

| Parameter | Value | Description |
|---|--|---|
| APPLICATION-STATISTICS Optional Scope: Global, Node | List: • YES • NO <i>Default: NO</i> | Indicates whether statistics collection is enabled for applications (including TELNET). |

Examples of Parameter Usage

The following are examples of TELNET configuration:

1. Node 1 initiates a 5-minute TELNET session to node 2, and begins the session at 50 simulation seconds into the total simulation time.

```
TELNET 1 2 5M 50S
```

2. Node 1 sends node 2 TELNET traffic for a duration randomly determined by TCPlib at the start of the simulation.

```
TELNET 1 2 0S 0S
```

8.8.3 GUI Configuration

TELNET is configured using the TELNET Properties Editor.

To configure a TELNET session between two nodes, perform the following steps:

1. Click the **TELNET** button in the **Applications** tab of the Standard Toolset.
2. On the canvas, click on the source node; drag the mouse to the destination node, and release.
3. Open the TELNET Properties Editor by doing one of the following:
 - a. On the canvas, either double-click on the application link or right-click on the application link and select **Properties** from the menu.
 - b. In the **Applications** tab of Table View, either double-click on the application row or right-click on the application row and select **Properties** from the menu.

4. Set the parameters listed in Table 8-66.

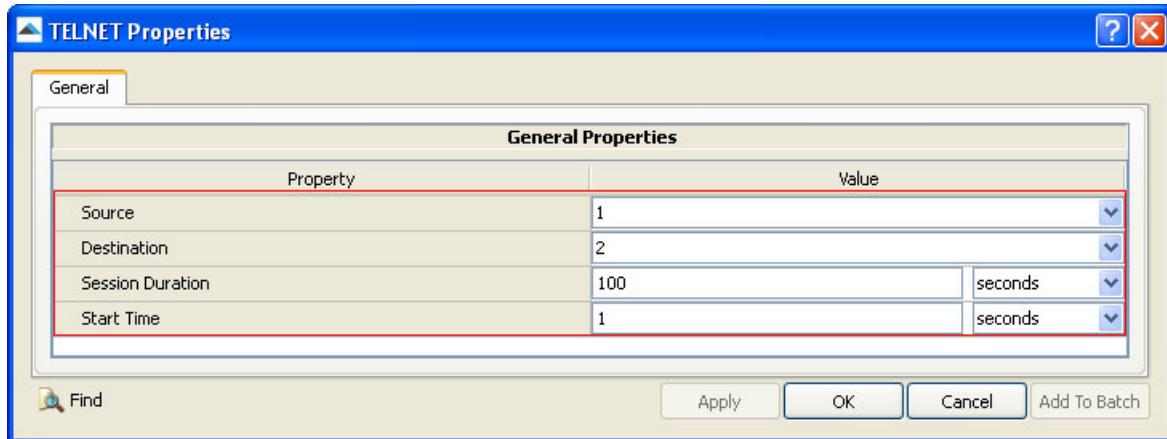


FIGURE 8-34. Setting TELNET Parameters

TABLE 8-66. Command Line Equivalent of TELNET Parameters

| GUI Parameter | Command Line Parameter |
|------------------|------------------------|
| Source | <src> |
| Destination | <dest> |
| Session Duration | <session-duration> |
| Start Time | <start-time> |

Setting Parameters

- To specify an IP address as the source (destination) ID, set **Source (Destination)** to one of the IP addresses listed in the drop-down list.

Configuring Statistics Parameters

Statistics for applications (including TELNET) can be collected at the global and node levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for TELNET, check the box labeled **Application** in the appropriate properties editor.

TABLE 8-67. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| Application | Global, Node | APPLICATION-STATISTICS |

8.8.4 Statistics

[Table 8-68](#) lists the TELNET statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 8-68. TELNET Statistics

| Statistics | Description |
|--------------------------|---|
| TELNET Client | |
| Server address | Specifies the server address |
| First Packet Sent at (s) | Specifies the time when the first packet is sent (seconds) |
| Last Packet Sent at (s) | Specifies the time when the last packet is sent (seconds) |
| Session Status | Client status of the session (open or closed) at the end of simulation |
| Total Bytes Sent | Specifies the total number of bytes sent |
| Total Bytes Received | Specifies the total number of bytes received |
| Throughput (bits/s) | Specifies the total throughput of the client (bits per second). See Note 1 below. |
| TELNET Server | |
| Client Address | Specifies the client address |
| First Packet Sent at (s) | Specifies the time when the first packet is sent (seconds) |
| Last Packet Sent at (s) | Specifies the time when the last packet is sent (seconds) |
| Session Status | Current server status of the session (open or closed) |
| Total Bytes Sent | Specifies the total number of bytes sent |
| Total Bytes Received | Specifies the total number of bytes received |
| Throughput (bits/s) | Specifies the total throughput of the server (bits per second). See note 2 below. |

Notes: 1. The throughput at the client is calculated as follows:

- If the session is complete, i.e., if all packets have been sent before the simulation ends, throughput = $(\text{total bytes sent} * 8) / (\text{time last packet sent} - \text{time first packet sent})$, where the times are in seconds.
- If the session is incomplete, i.e., if all packets have not been sent before the simulation ends, throughput = $(\text{total bytes sent} * 8) / (\text{simulation time} - \text{time first packet sent})$, where the times are in seconds.

2. The throughput at the server is calculated as follows:

- If the session is complete, throughput = $(\text{total bytes received} * 8) / (\text{time last packet received} - \text{time first packet received})$, where the times are in seconds.
- If the session is incomplete, throughput = $(\text{total bytes received} * 8) / (\text{simulation time} - \text{time first packet received})$, where the times are in seconds.

8.9 Traffic Generator (Traffic-Gen)

8.9.1 Description

The Traffic Generator model simulates a random-distribution based traffic generator. These random distributions are applicable to session property and traffic property. Also, the application is capable of shaping traffic by using either leaky bucket or dual leaky bucket.

8.9.2 Command Line Configuration

Configuration Requirements

In order to configure a Traffic-Gen session with QoS constraints, a path must exist from the source to each destination node along which QOSPF (Quality of Service Extensions to OSPF) is enabled. See *Multimedia and Enterprise Model Library* for details of configuring QOSPF.

Application Configuration File Parameters

To specify Traffic-Gen traffic, include the following statement in the application configuration (.app) file:

```
TRAFFIC-GEN <Source> <Destination> <Session Parameters>
    <Traffic Parameters> <Leaky Bucket Parameters>
    [<Fragmentation Parameters>] [<QoS Parameters>]
    [<MDP Parameters>] [App Name Parameters]
```

Note: All parameters should be entered on the same line.

The Traffic-Gen parameters are described in [Table 8-69](#).

TABLE 8-69. Traffic-Gen Parameters

| Element | Description | | | | | | | | |
|---------------------------|--|---------------|--|-------------------|--|---------------|---|----------|---|
| <Source> | Node ID or IP address of the source node. | | | | | | | | |
| <Destination> | Node ID or IP address of the destination node or a multicast address. | | | | | | | | |
| <Session Parameters> | <p>Session properties (start time and duration) of the connection. The session properties are specified in the following format:</p> <pre><Start Time> <Duration></pre> <p>where</p> <table> <tr> <td><Start Time></td> <td>Time when the session starts. This is specified as a time distribution (see note 1).</td> </tr> <tr> <td><Duration></td> <td>Length of the session. This is specified as a time distribution (see note 1).</td> </tr> </table> | <Start Time> | Time when the session starts. This is specified as a time distribution (see note 1). | <Duration> | Length of the session. This is specified as a time distribution (see note 1). | | | | |
| <Start Time> | Time when the session starts. This is specified as a time distribution (see note 1). | | | | | | | | |
| <Duration> | Length of the session. This is specified as a time distribution (see note 1). | | | | | | | | |
| <Traffic Parameters> | <p>Traffic properties (size and inter-arrival times of data elements) of the connection. The traffic properties are specified in the following format:</p> <pre>RND <Packet Size> <Packet Interval> <Probability></pre> <p>where</p> <table> <tr> <td><Packet Size></td> <td>Packet size, in bytes. This is specified as an integer distribution (see note 1).</td> </tr> <tr> <td><Packet Interval></td> <td>Time between consecutive packets. This is specified as a time distribution (see note 1).</td> </tr> <tr> <td><Probability></td> <td>Probability of sending a packet. This is a real number in the range [0.0, 1.0].</td> </tr> </table> | <Packet Size> | Packet size, in bytes. This is specified as an integer distribution (see note 1). | <Packet Interval> | Time between consecutive packets. This is specified as a time distribution (see note 1). | <Probability> | Probability of sending a packet. This is a real number in the range [0.0, 1.0]. | | |
| <Packet Size> | Packet size, in bytes. This is specified as an integer distribution (see note 1). | | | | | | | | |
| <Packet Interval> | Time between consecutive packets. This is specified as a time distribution (see note 1). | | | | | | | | |
| <Probability> | Probability of sending a packet. This is a real number in the range [0.0, 1.0]. | | | | | | | | |
| <Leaky Bucket Parameters> | <p>Leaky bucket properties for traffic shaping. There are three options of specifying leaky bucket properties.</p> <p>If traffic shaping is not used, use the following format:</p> <pre>NLB</pre> <p>If leaky bucket is used, for traffic shaping use the following format:</p> <pre>LB <Bucket Size> <Token Rate> <Action></pre> <p>If dual leaky bucket is used for traffic shaping, use the following format:</p> <pre>DLB <Bucket Size> <Token Rate> <Peak Rate> <Action></pre> <p>where</p> <table> <tr> <td><Bucket Size></td> <td>Bucket size, in bytes.</td> </tr> <tr> <td><Token Rate></td> <td>Token generation rate, in bps.</td> </tr> <tr> <td><Peak Rate></td> <td>Maximum allowed rate, in bps.</td> </tr> <tr> <td><Action></td> <td>Action to perform. This can be DROP or DELAY.</td> </tr> </table> | <Bucket Size> | Bucket size, in bytes. | <Token Rate> | Token generation rate, in bps. | <Peak Rate> | Maximum allowed rate, in bps. | <Action> | Action to perform. This can be DROP or DELAY. |
| <Bucket Size> | Bucket size, in bytes. | | | | | | | | |
| <Token Rate> | Token generation rate, in bps. | | | | | | | | |
| <Peak Rate> | Maximum allowed rate, in bps. | | | | | | | | |
| <Action> | Action to perform. This can be DROP or DELAY. | | | | | | | | |

TABLE 8-69. Traffic-Gen Parameters (Continued)

| Element | Description | | | | | | | | | | |
|----------------------------|--|-----------------|--|---------|-----------------------------------|------------|--|------------------|--|------------|--|
| <Fragmentation Parameters> | <p>Fragment size specification.</p> <p>Note: Fragment size specification is optional.</p> <p>The fragment size is specified in the following format:</p> <pre>FRAGMENT-SIZE <Fragment-size></pre> <p>where</p> <table> <tr> <td><Fragment-size></td> <td>Size of fragments, in bytes.</td> </tr> </table> | <Fragment-size> | Size of fragments, in bytes. | | | | | | | | |
| <Fragment-size> | Size of fragments, in bytes. | | | | | | | | | | |
| <QoS Parameters> | <p>QoS properties of the connection.</p> <p>Note: QoS properties specification is optional can only be included for client-server sessions.</p> <p>The QoS properties are specified in the following format:</p> <pre>CONSTRAINT <Bandwidth> <Delay> [<Priority>] [<Retry Property>]</pre> <p>where</p> <table> <tr> <td><Bandwidth></td> <td>QoS Bandwidth, in bps.</td> </tr> <tr> <td><Delay></td> <td>QoS end-to-end delay, in seconds.</td> </tr> <tr> <td><Priority></td> <td>Priority given to a session. See note 2.</td> </tr> <tr> <td><Retry Property></td> <td> <p>Note: Priority specification is optional.</p> <p>Retry property, which is specified in the following format:</p> <pre>RETRY-INTERVAL <interval></pre> <p>where</p> <table> <tr> <td><interval></td> <td>Retry interval, specified as a time value.</td> </tr> </table> </td> </tr> </table> <p>Note: Retry property specification is optional.</p> | <Bandwidth> | QoS Bandwidth, in bps. | <Delay> | QoS end-to-end delay, in seconds. | <Priority> | Priority given to a session. See note 2. | <Retry Property> | <p>Note: Priority specification is optional.</p> <p>Retry property, which is specified in the following format:</p> <pre>RETRY-INTERVAL <interval></pre> <p>where</p> <table> <tr> <td><interval></td> <td>Retry interval, specified as a time value.</td> </tr> </table> | <interval> | Retry interval, specified as a time value. |
| <Bandwidth> | QoS Bandwidth, in bps. | | | | | | | | | | |
| <Delay> | QoS end-to-end delay, in seconds. | | | | | | | | | | |
| <Priority> | Priority given to a session. See note 2. | | | | | | | | | | |
| <Retry Property> | <p>Note: Priority specification is optional.</p> <p>Retry property, which is specified in the following format:</p> <pre>RETRY-INTERVAL <interval></pre> <p>where</p> <table> <tr> <td><interval></td> <td>Retry interval, specified as a time value.</td> </tr> </table> | <interval> | Retry interval, specified as a time value. | | | | | | | | |
| <interval> | Retry interval, specified as a time value. | | | | | | | | | | |

TABLE 8-69. Traffic-Gen Parameters (Continued)

| Element | Description | | | | | | |
|--|---|---|--|--|--|---|--|
| <p><MDP Parameters></p> <p>MDP parameters of the connection.</p> <p>Note: MDP parameter specification is optional. If MDP parameters are not specified, the application does not run with MDP.</p> <p>The MDP parameters are specified in the following format:</p> <pre>MDP-ENABLED [MDP-PROFILE <profile-name>]</pre> <p>where</p> <table> <tr> <td style="vertical-align: top;"> <p><profile-name></p> </td><td>Name of the MDP profile to be used with the application.</td></tr> <tr> <td></td><td>This should be the name of a MDP profile defined in the MDP profile file (see Section 7.2.4.3).</td></tr> <tr> <td></td><td>This MDP profile takes precedence over the MDP profile associated with the node in the scenario configuration file, if any.</td></tr> </table> <p>Note: If parameter MDP-ENABLED is included but parameter MDP-PROFILE is not included, and a MDP profile is specified for the node in the scenario configuration file, then the MDP profile specified for the node is used.</p> <p>If parameter MDP-ENABLED is included but parameter MDP-PROFILE is not included, and a MDP profile is not specified for the node in the scenario configuration file, then default MDP values for the MDP profile parameters are used (see Section 7.2.4.3).</p> | <p><profile-name></p> | Name of the MDP profile to be used with the application. | | This should be the name of a MDP profile defined in the MDP profile file (see Section 7.2.4.3). | | This MDP profile takes precedence over the MDP profile associated with the node in the scenario configuration file, if any. | |
| <p><profile-name></p> | Name of the MDP profile to be used with the application. | | | | | | |
| | This should be the name of a MDP profile defined in the MDP profile file (see Section 7.2.4.3). | | | | | | |
| | This MDP profile takes precedence over the MDP profile associated with the node in the scenario configuration file, if any. | | | | | | |
| <p><App Name Parameters></p> <p>Session name associated with the connection.</p> <p>Note: Session name specification is optional.</p> <p>The session name is specified in the following format:</p> <pre>APPLICATION-NAME <application-name></pre> <p>where</p> <table> <tr> <td style="vertical-align: top;"> <p><application-name></p> </td><td>Name of the Traffic-Gen session. This name is printed in the statistics file and statistics database.</td></tr> </table> | <p><application-name></p> | Name of the Traffic-Gen session. This name is printed in the statistics file and statistics database. | | | | | |
| <p><application-name></p> | Name of the Traffic-Gen session. This name is printed in the statistics file and statistics database. | | | | | | |

Notes: 1. Integer and Time Distributions: Several Traffic-Gen parameters are specified as random number distributions. Five random number distributions are supported: deterministic, uniform, exponential, truncated Pareto, and 4-parameter truncated Pareto.

- The deterministic distribution is specified as:

```
DET <det-val>
```

It always returns `<det-val>` as the value.

- The uniform distribution is specified as:

```
UNI <uni-val-1> <uni-val-2>
```

It returns a value uniformly distributed between `<uni-val-1>` and `<uni-val-2>`.

- The exponential distribution is specified as:

```
EXP <exp-val>
```

It returns a value from an exponential distribution with `<exp-val>` as the mean.

- The truncated Pareto distribution is specified as:

```
TPD <tp-val-1> <tp-val-2> <tp-alpha>
```

It returns a value from a truncated Pareto distribution with `<tp-val-1>` as the lower end of the range, `<tp-val-2>` as the upper limit of the truncation, and `<tp-alpha>` as the shape parameter.

- The 4-parameter truncated Pareto distribution is specified as:

```
TPD4 <tp4-val-1> <tp-val-2> <tp4-val-3> <tp4-alpha>
```

It returns a value from a truncated Pareto distribution with `<tp4-val-1>` as the lower end of the range, `<tp4-val-2>` as the lower limit of the truncation, `<tp4-val-3>` as the upper limit of the truncation, and `<tp4-alpha>` as the shape parameter.

For integer distributions, `<det-val>`, `<uni-val-1>`, `<uni-val-2>`, `<exp-val>`, `<tp-val-1>`, `<tp-val-2>`, `<tp-alpha>`, `<tp4-val-1>`, `<tp4-val-2>`, `<tp4-val-3>`, and `<tp4-alpha>` are integer values, e.g., 0, 10, 15, etc.

For time distributions, `<det-val>`, `<uni-val-1>`, `<uni-val-2>`, `<exp-val>`, `<tp-val-1>`, `<tp-val-2>`, `<tp-alpha>`, `<tp4-val-1>`, `<tp4-val-2>`, `<tp4-val-3>`, and `<tp4-alpha>` are time values, e.g., 5S, 0.5MS, 100US, etc.

2. Priority Specification: Priority can be specified by including a TOS specification, DSCP specification, or Precedence specification.

- TOS specification has the following format:

```
TOS <TOS-value>
```

where `<TOS-value>` is the value of the TOS bits of the IP header. `<TOS-value>` should be an integer in the range [0, 255].

- DSCP specification has the following format:

```
DSCP <DSCP-value>
```

where `<DSCP-value>` is the value of the DSCP bits of the IP header. `<DSCP-value>` should be an integer in the range [0, 63].

- Precedence specification has the following format:

```
PRECEDENCE <precedence-value>
```

where `<precedence-value>` is the value of the Precedence bits of the IP header. `<precedence-value>` should be an integer in the range [0, 7].

At most one of the three parameters TOS, DSCP, or PRECEDENCE can be specified. If a TOS specification, DSCP specification, or Precedence specification is not included, PRECEDENCE 0 is used as default.

Scenario Configuration File Parameters

[Table 8-70](#) describes the Traffic-Gen parameters that can be specified in the scenario configuration (.config) file.

TABLE 8-70. Traffic-Gen Scenario Configuration File Parameters

| Parameter | Value | Description |
|--|--|---|
| APPLICATION-STATISTICS <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none"> • YES • NO <i>Default:</i> NO | Indicates whether statistics collection is enabled for applications (including Traffic-Gen). |
| TRACE-TRAFFIC-GEN <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none"> • YES • NO <i>Default:</i> YES | Indicates whether packet tracing is enabled for Traffic-Gen. Note: To enable packet tracing, some other parameters need to be configured as well. Refer to Section 4.2.10 of <i>QualNet User's Guide</i> for details. |

Examples of Parameter Usage

The following are examples of Traffic-Gen configuration:

1. This is an example of voice traffic that can be modeled with a simple two-state Markov chain. The chain consists of an ON state of exponentially distributed duration with an average of 352 ms, and an OFF state of 648 ms. In addition, the voice application uses PCM encoding which generates 160-bytes voice data every 20 ms.

```
TRAFFIC-GEN 1 2 DET 0S DET 600S RND DET 160 DET 20MS 0.352 NOLB
```

This line specifies traffic from node 1 to node 2, beginning at time 0, with a duration of 600 seconds. The traffic is random (as opposed to file-based) with a fixed size of 160 bytes and a fixed interval of 20 milliseconds. The probability field is a value between 0 and 1 used to determine what percent of the packets described by the given pattern will be sent. This is to model the ON and OFF states (in a somewhat less than ideal way). In this case, only about 35% of the packets will be sent. If probability is set to 1.0, the traffic is essentially identical to the CBR traffic generated by CBR 1 2 0 160 20MS 0S 600S. The final parameter NOLB means that the traffic is not shaped by leaky bucket.

[Figure 8-35](#) displays the result of this voice model. The throughput of the voice traffic and the overall average are shown. The overall average can be verified by 64 Kbps (= 160 bytes / 20 ms) * 0.352 = 23 Kbps.

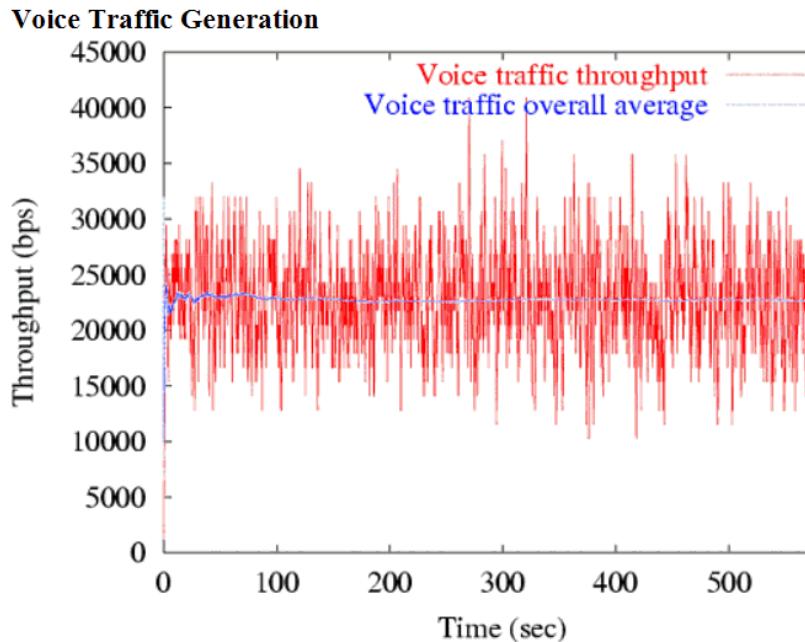


FIGURE 8-35. Voice Traffic Generation

2. This is an example of an arbitrary traffic model in which the data length is exponentially distributed with an average of 160 bytes, and the inter-arrival time is also exponentially distributed with an average of 20 ms, and the data generation probability is 1. There is no traffic shaping.

```
TRAFFIC-GEN 1 2 DET 0 DET 600 RND EXP 160 EXP 20MS 1 NOLB
```

The overall average rate of this traffic can be computed as:

$$\text{avg} = 160 \text{ bytes} / 20 \text{ ms} = 64 \text{ Kbps}$$

This average rate is displayed in the following Random Traffic Generation graphic.

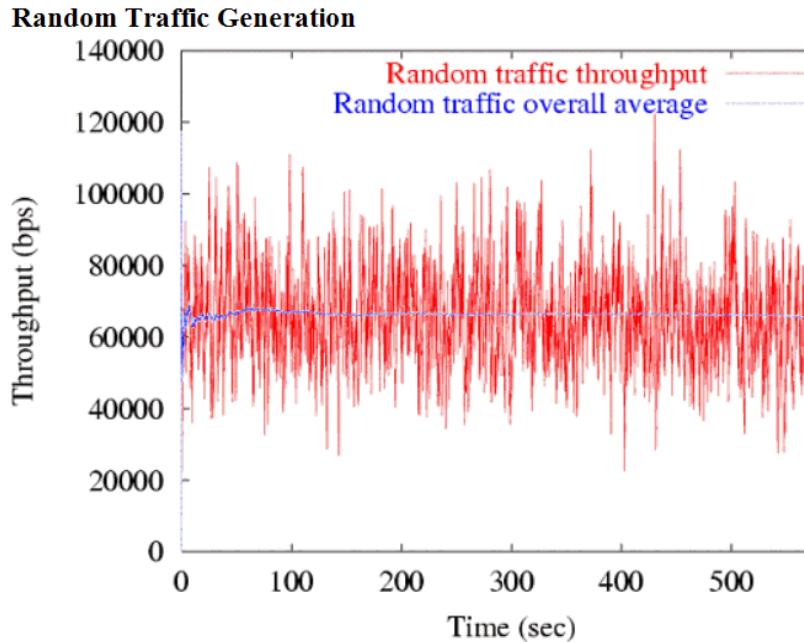


FIGURE 8-36. Random Traffic Generation

3. This is the same as the previous example, except that Traffic-Gen runs with MDP and uses the MDP profile associated with node 1. If there is no MDP profile associated with node 1, default values for MDP profile parameters are used.

```
TRAFFIC-GEN 1 2 DET 0 DET 600 RND EXP 160 EXP 20MS 1 NOLB MDP-ENABLED
```

4. This is the same as the previous example, except that MDP uses the user-defined MDP profile profile-1.

```
TRAFFIC-GEN 1 2 DET 0 DET 600 RND EXP 160 EXP 20MS 1 NOLB MDP-ENABLED  
MDP-PROFILE profile-1
```

8.9.3 GUI Configuration

[Section 8.9.3.1](#) describes how to configure a Traffic-Gen session between two nodes. [Section 8.9.3.2](#) describes how to configure a single host Traffic-Gen session. [Section 8.9.3.3](#) describes how to configure statistics parameters for applications (including Traffic-Gen).

Configuration Requirements

In order to configure a Traffic-Gen session with QoS constraints, a path must exist from the source to each destination node along which QOSPF (Quality of Service Extensions to OSPF) is enabled. See *Multimedia and Enterprise Model Library* for details of configuring QOSPF.

8.9.3.1 Configuring a Client-Server Traffic-Gen Session

To configure a Traffic-Gen session between two nodes, perform the following steps:

1. Click the **TRAF GEN** button in the **Applications** tab of the Standard Toolset.
2. On the canvas, click on the source node, drag the mouse to the destination node, and release.
3. Open the Traffic Generator Properties Editor by doing one of the following:
 - Right-click in the application link on the canvas and select **Properties** from the menu.
 - In the **Applications** tab of Table View either double-click on the application row or right-click on the application row and select **Properties** from the menu.
4. Set the application parameters as described in [Section 8.9.3.1.1](#) to [Section 8.9.3.1.7](#).
5. Set the statistics parameters as described in [Section 8.9.3.3](#).

8.9.3.1.1 Configuring Source, Destination, and Session Name Parameters

Set the source, destination and session name parameters listed in [Table 8-69](#).

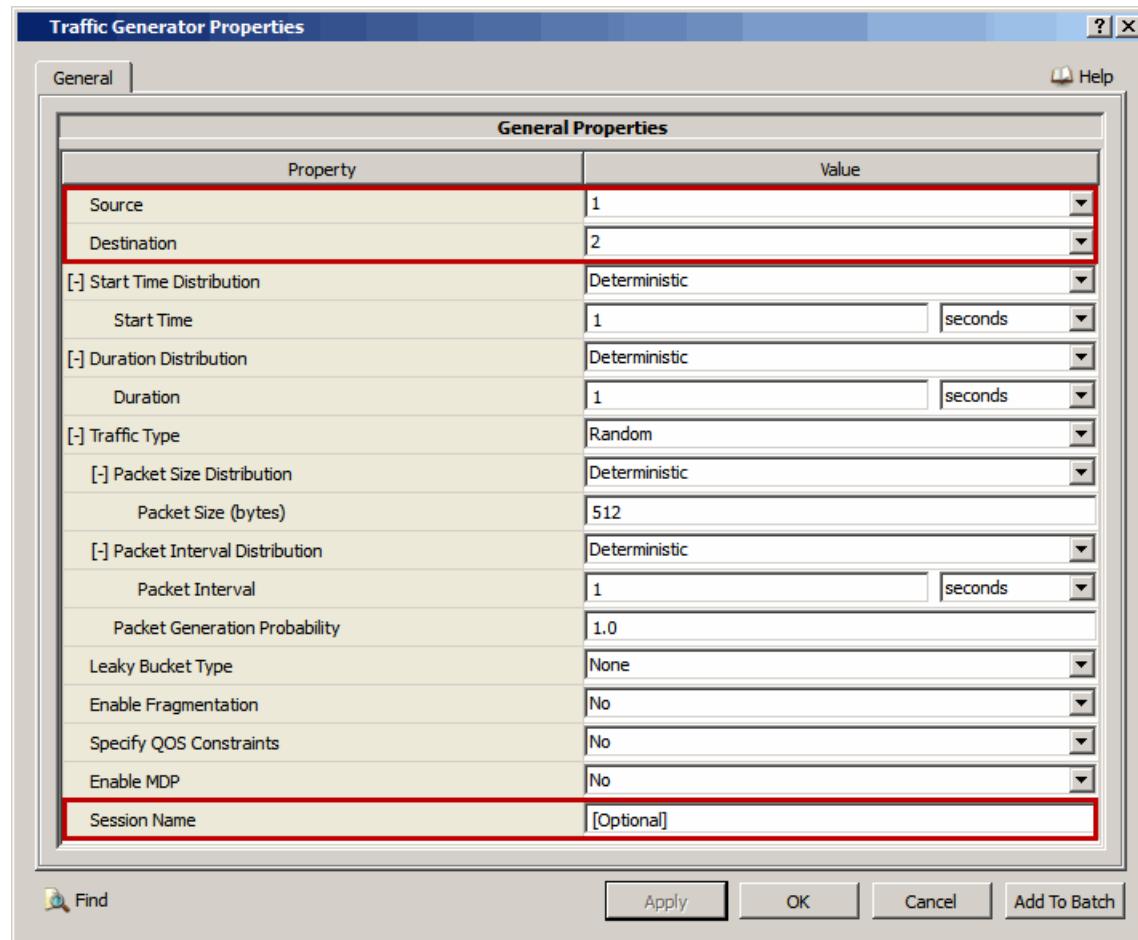


FIGURE 8-37. Setting Source, Destination, Session Name and Parameters

TABLE 8-71. Command Line Equivalent of Source, Destination, and Session Name Parameters

| GUI Parameter | Command Line Parameter |
|---------------|-------------------------------------|
| Source | <Source> |
| Destination | <Destination> |
| Session Name | APPLICATION-NAME <application-name> |

Setting Parameters

- To specify an IP address as the source (destination) ID, set **Source (Destination)** to one of the IP addresses listed in the drop-down list.

8.9.3.1.2 Configuring Session Parameters

To configure the session parameters, perform the following steps:

1. Set the parameters **Start Time Distribution** and **Duration Distribution**.

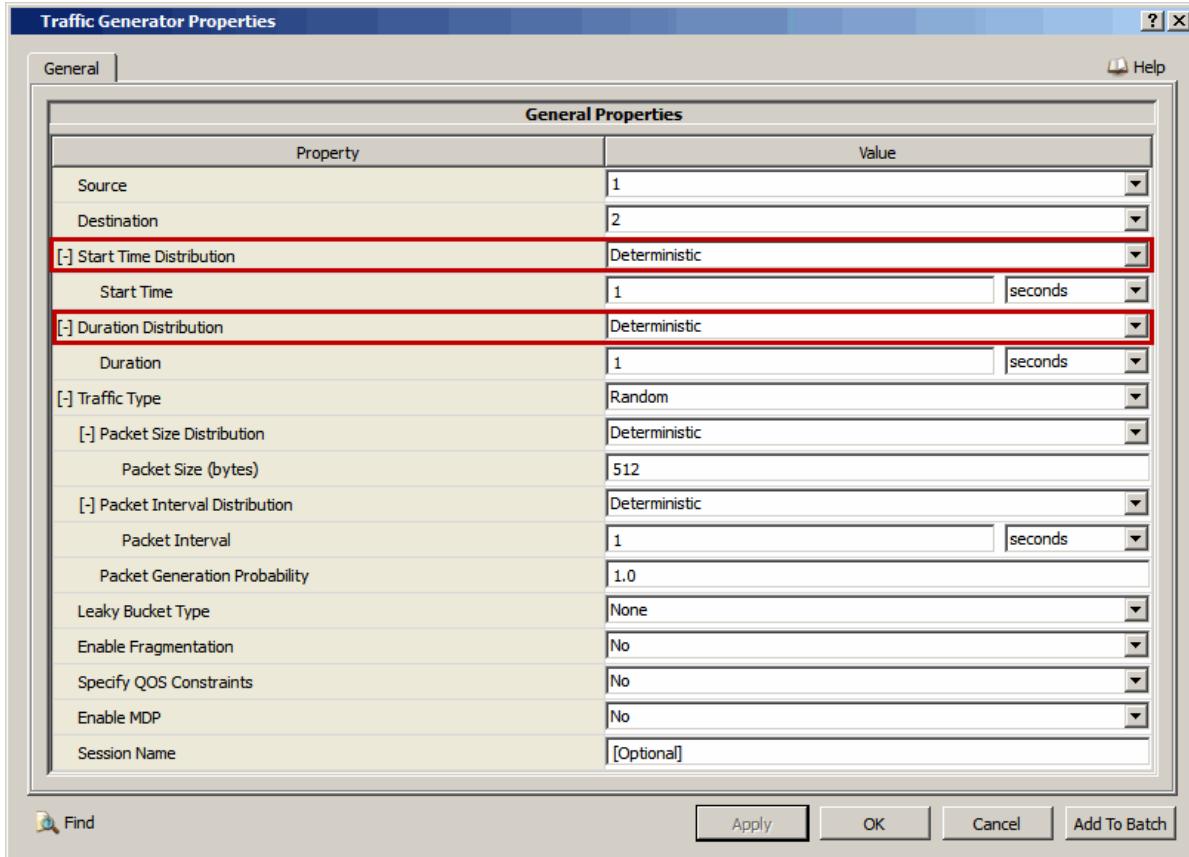


FIGURE 8-38. Setting Session Parameters

TABLE 8-72. Command Line Equivalent of Session Parameters

| GUI Parameter | Command Line Parameter |
|---|------------------------|
| Start Time Distribution and dependent parameters | <Start Time> |
| Duration Time Distribution and dependent parameters | <Duration> |

2. For each of the parameters **Start Time Distribution** and **Duration Distribution**, set the dependent parameters of the selected distribution. We show below how to set the dependent parameters for **Start Time Distribution** for each applicable distribution. Dependent parameters for **Duration Distribution** can be set in a similar way.

- If **Start Time Distribution** is set to *Deterministic*, then set the dependent parameters listed in Table 8-73.

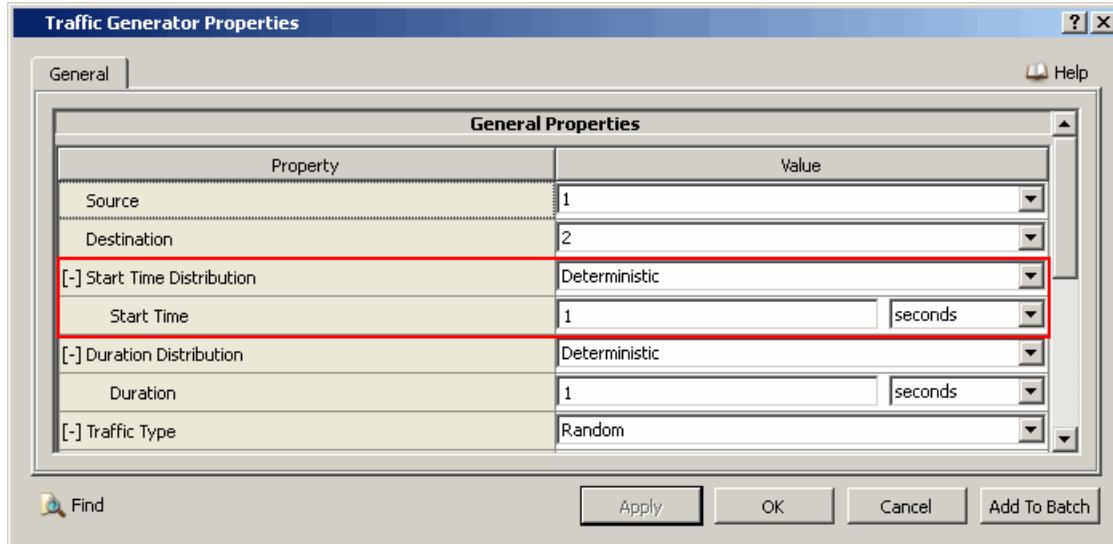


FIGURE 8-39. Setting Parameters for a Deterministic Distribution

TABLE 8-73. Command Line Equivalent of Deterministic Distribution Parameters

| GUI Parameter | Command Line Parameter |
|--|------------------------|
| Start Time Distribution (set to <i>Deterministic</i>) | DET |
| Start Time | <det-val> |

- If **Start Time Distribution** is set to *Exponential*, then set the dependent parameters listed in Table 8-74.

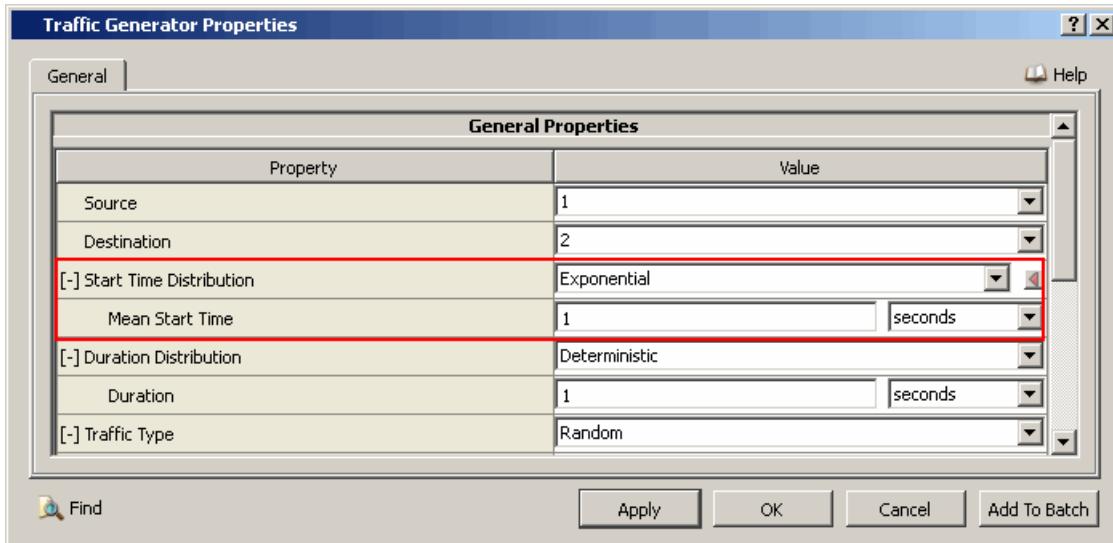
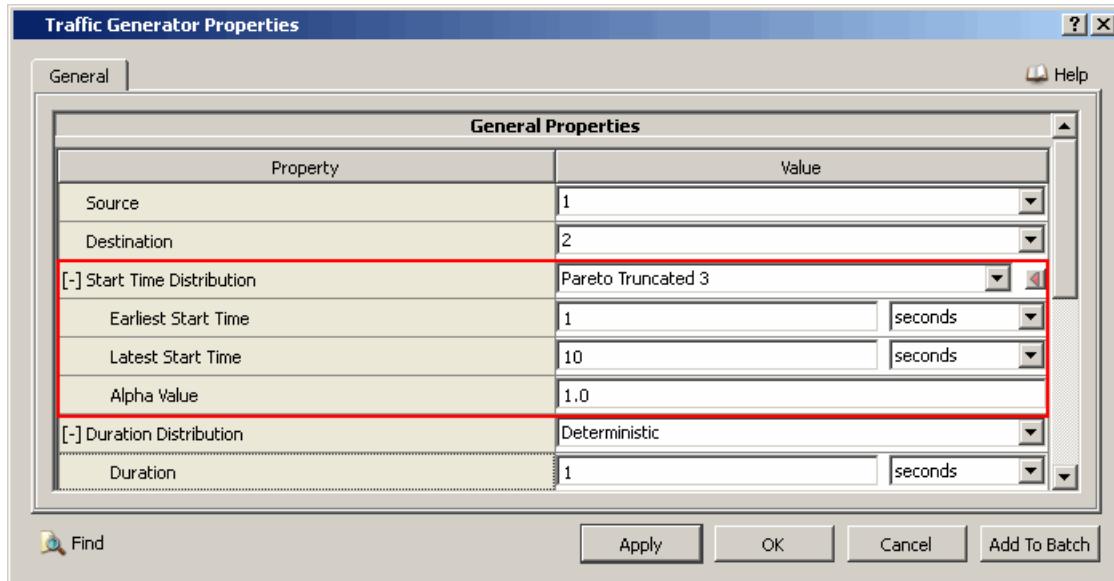


FIGURE 8-40. Setting Parameters for an Exponential Distribution

TABLE 8-74. Command Line Equivalent of Exponential Distribution Parameters

| GUI Parameter | Command Line Parameter |
|--|------------------------|
| Start Time Distribution (set to <i>Exponential</i>) | EXP |
| Mean Start Time | <exp-val> |

- If **Start Time Distribution** is set to *Pareto Truncated 3*, then set the dependent parameters listed in Table 8-75.

**FIGURE 8-41.** Setting Parameters for a Truncated Pareto Distribution**TABLE 8-75.** Command Line Equivalent of Truncated Pareto Distribution Parameters

| GUI Parameter | Command Line Parameter |
|---|------------------------|
| Start Time Distribution (set to <i>Pareto Truncated 3</i>) | TPD |
| Earliest Start Time | <tp-val-1> |
| Latest Start Time | <tp-val-2> |
| Alpha Value | <tp-alpha> |

- If **Start Time Distribution** is set to *Pareto Truncated 4*, then set the dependent parameters listed in Table 8-76.

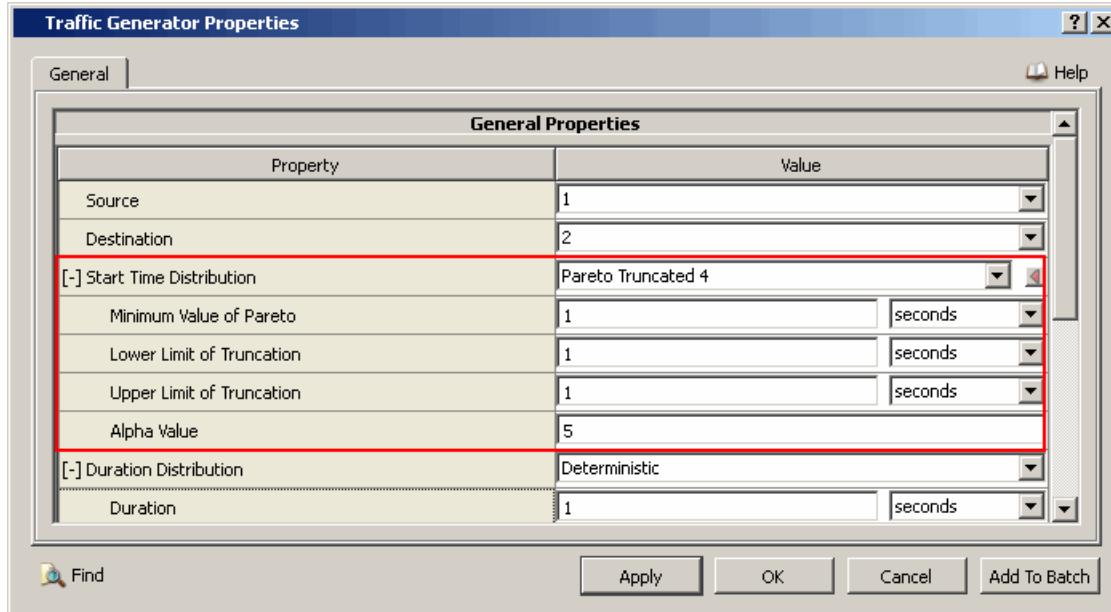


FIGURE 8-42. Setting Parameters for a 4-Parameter Truncated Pareto Distribution

TABLE 8-76. Command Line Equivalent of 4-Parameter Truncated Pareto Distribution Parameters

| GUI Parameter | Command Line Parameter |
|---|------------------------|
| Start Time Distribution (set to <i>Pareto Truncated 4</i>) | TPD4 |
| Minimum Value of Pareto | <tp4-val-1> |
| Lower Limit of Truncation | <tp4-val-2> |
| Upper Limit of Truncation | <tp4-val-3> |
| Alpha Value | <tp4-alpha> |

- If **Start Time Distribution** is set to *Uniform* then set the dependent parameters listed in [Table 8-77](#).

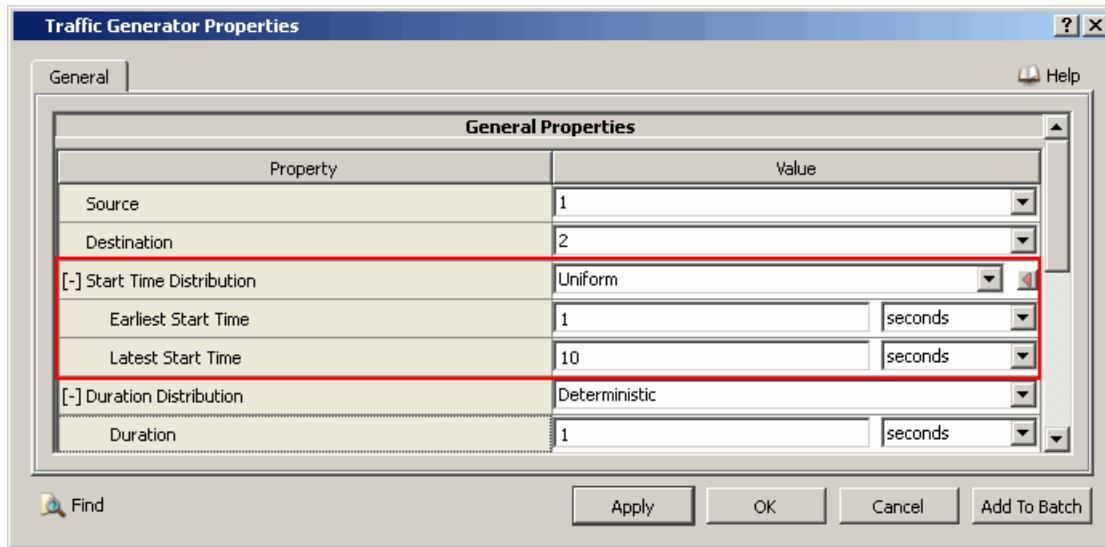


FIGURE 8-43. Setting Parameters for a Uniform Distribution

TABLE 8-77. Command Line Equivalent of Uniform Distribution Parameters

| GUI Parameter | Command Line Parameter |
|--|------------------------|
| Start Time Distribution (set to <i>Uniform</i>) | UNI |
| Earliest Start Time | <uni-val-1> |
| Latest Start Time | <uni-val-2> |

8.9.3.1.3 Configuring Traffic Parameters

To configure the traffic parameters, perform the following steps:

1. Set the parameters **Packet Size Distribution**, **Packet Interval Distribution**, and **Packet Generation Probability**.

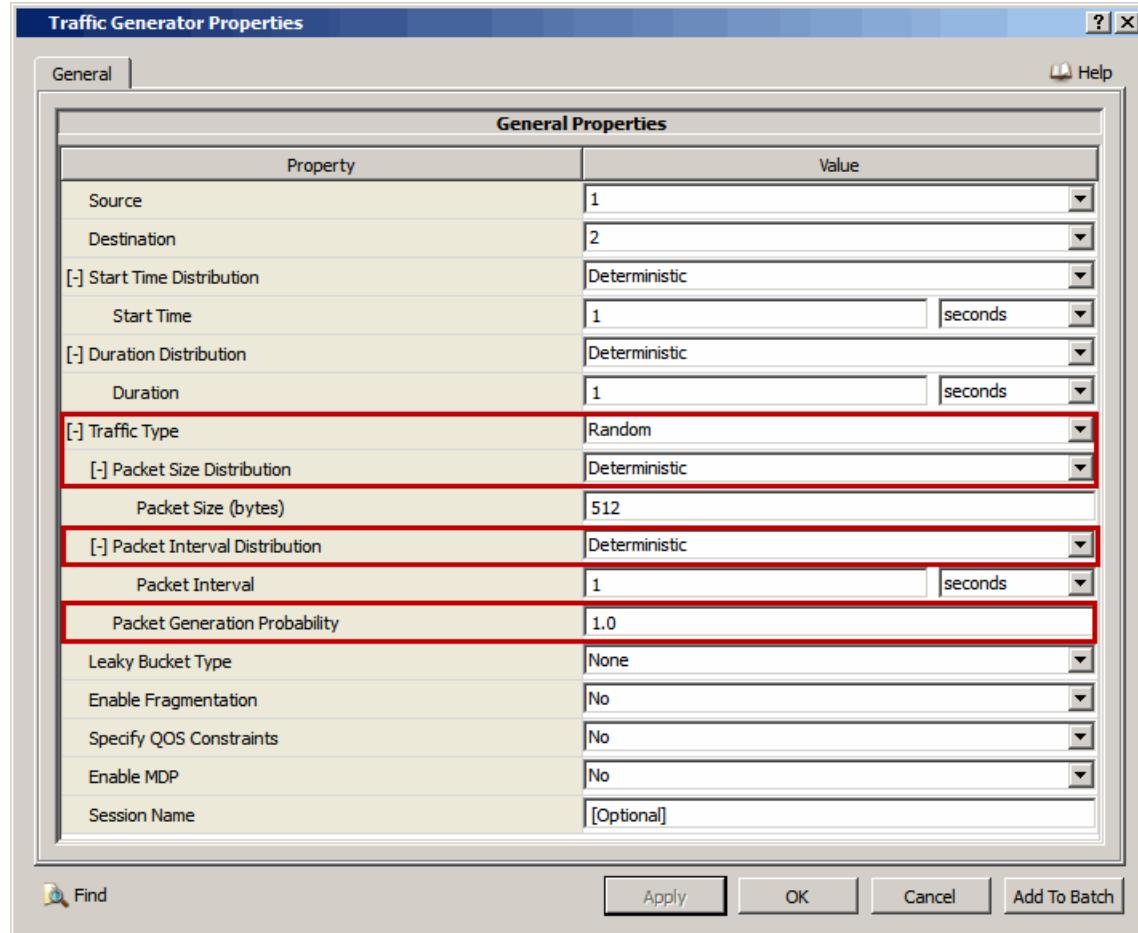


FIGURE 8-44. Setting Traffic Parameters

TABLE 8-78. Command Line Equivalent of Traffic Parameters

| GUI Parameter | Command Line Parameter |
|---|------------------------|
| Traffic Type (set to <i>Random</i>) | RND |
| Packet Size Distribution and dependent parameters | <Packet Size> |
| Packet Interval Distribution and dependent parameters | <Packet Interval> |
| Packet Generation Probability | <Probability> |

2. For each of the parameters **Packet Size Distribution** and **Packet Interval Distribution**, set the dependent parameters of the selected distribution. These parameters are set in the same way as the dependent parameters for **Start Time Distribution** (see [Section 8.9.3.1.2](#)).

8.9.3.1.4 Configuring Leaky Bucket Parameters

To configure the leaky bucket parameters, perform the following steps:

1. Set the parameter **Leaky Bucket Type**.

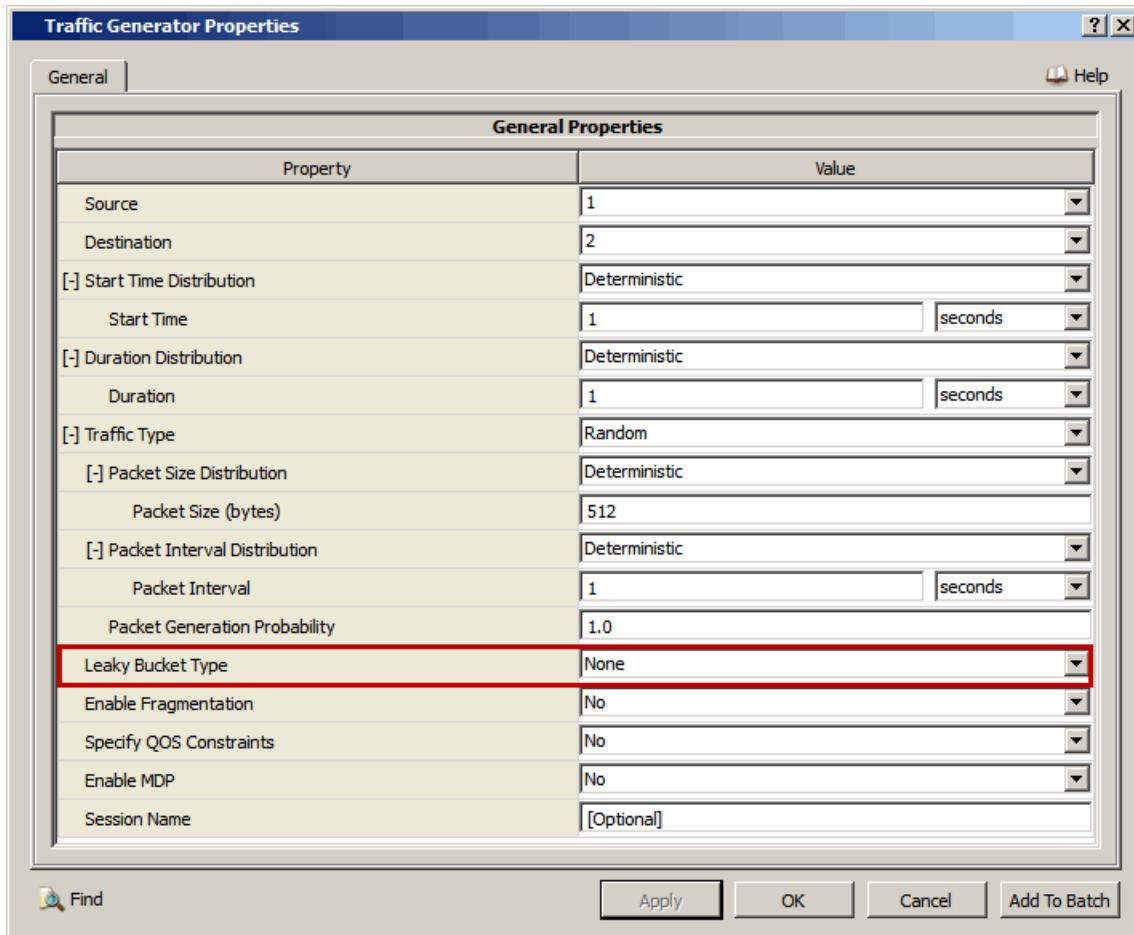


FIGURE 8-45. Configuring Leaky Bucket Parameters

TABLE 8-79. Command Line Equivalent of Leaky Bucket Parameters

| GUI Parameter | Command Line Parameter |
|--|---------------------------|
| Leaky Bucket Type and dependent parameters | <Leaky Bucket Parameters> |

Setting Parameters

- To use leaky bucket for traffic shaping, set **Leaky Bucket** to *Leaky Bucket*. To use dual leaky bucket for traffic shaping, set **Leaky Bucket** to *Dual Leaky Bucket*.

2. If **Leaky Bucket** is set to *None* then traffic shaping is not used and there are no other parameters to set.

TABLE 8-80. Command Line Equivalent of Leaky Bucket Parameters (Leaky Bucket Type = None)

| GUI Parameter | Command Line Parameter |
|---|------------------------|
| Leaky Bucket Type (set to <i>None</i>) | NLB |

3. If **Leaky Bucket** is set to *Leaky Bucket* then set the dependent parameters listed in [Table 8-81](#).

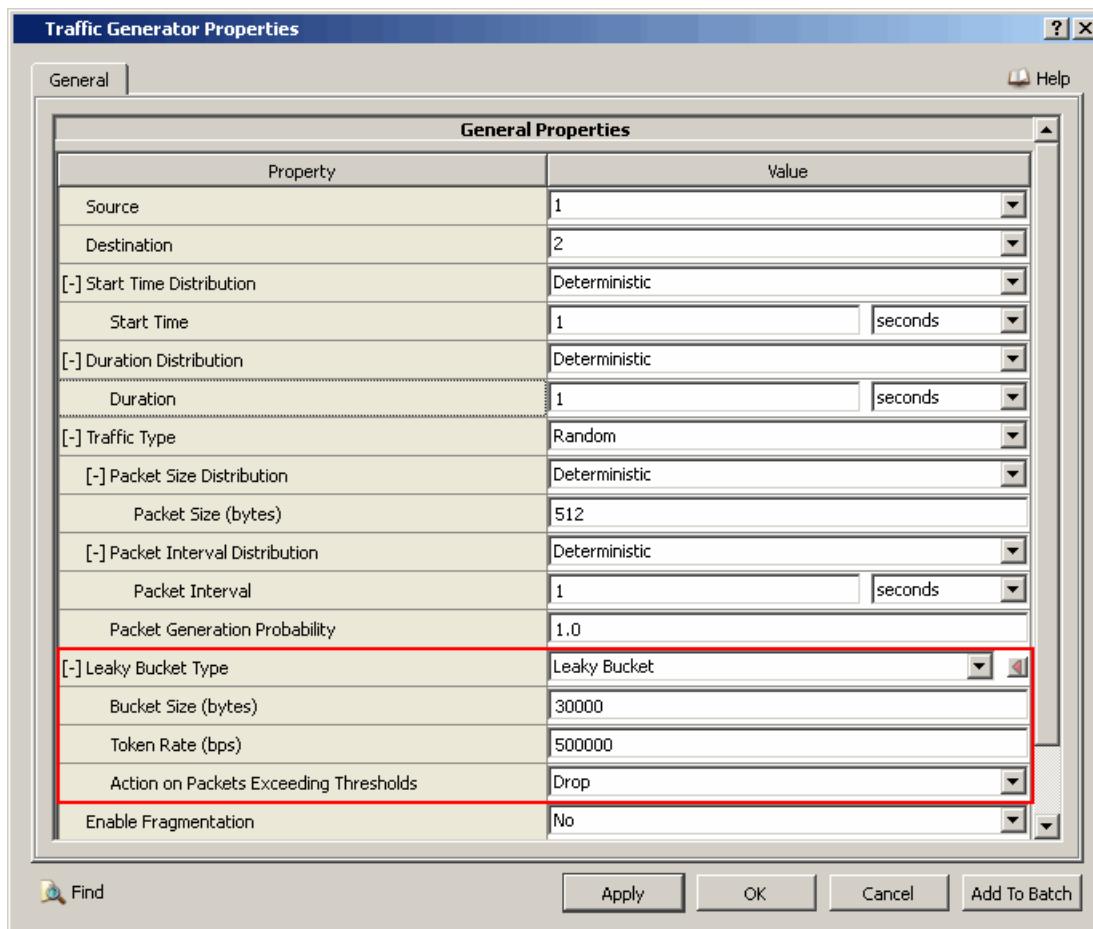


FIGURE 8-46. Setting Leaky Bucket Parameters

TABLE 8-81. Command Line Equivalent of Leaky Bucket Parameters (Leaky Bucket Type = Leaky Bucket)

| GUI Parameter | Command Line Parameter |
|---|------------------------|
| Leaky Bucket Type (set to <i>Leaky Bucket</i>) | LB |
| Bucket Size | <Bucket Size> |
| Token Rate | <Token Rate> |
| Action on Packets Exceeding Thresholds | <Action> |

4. If **Leaky Bucket** is set to *Dual Leaky Bucket* then set the dependent parameters listed in [Table 8-82](#).

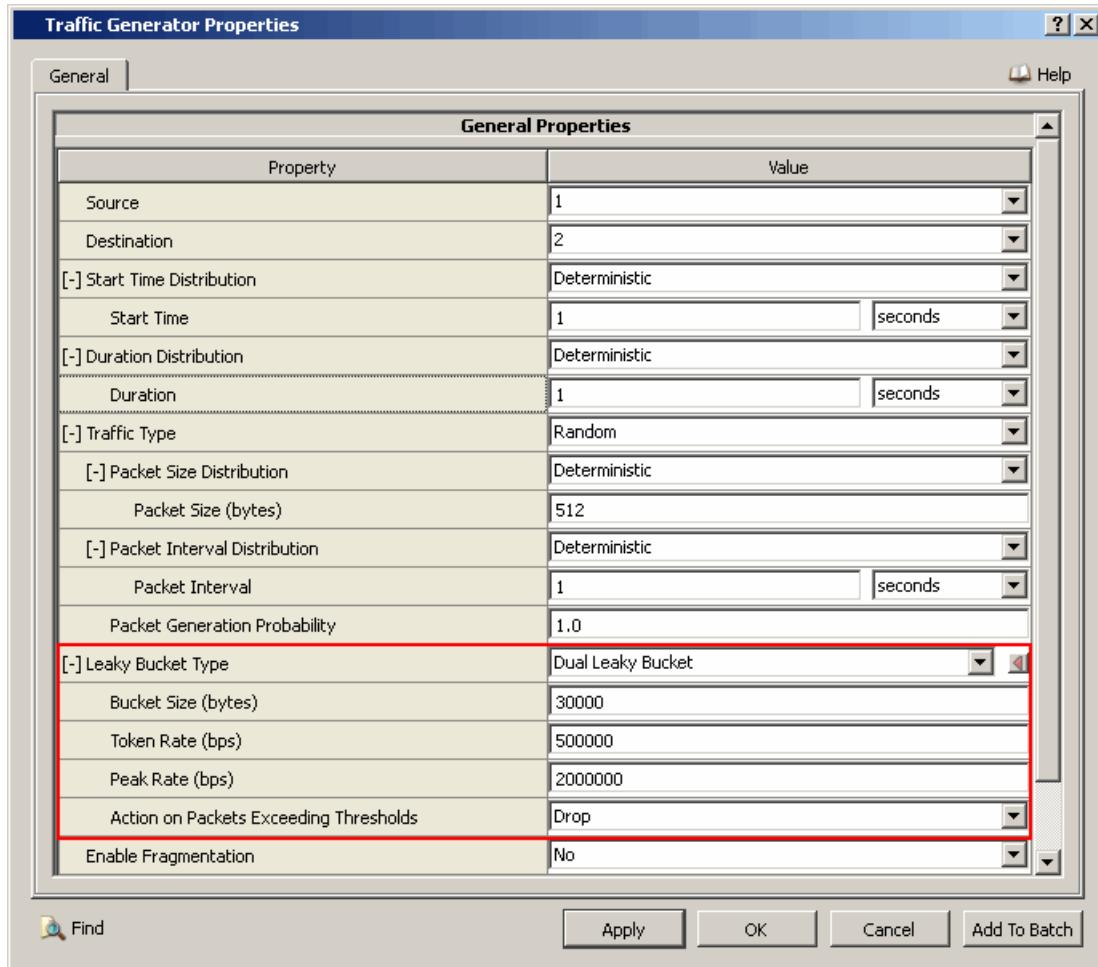


FIGURE 8-47. Setting Dual Leaky Bucket Parameters

TABLE 8-82. Command Line Equivalent of Leaky Bucket Parameters (Leaky Bucket Type = Dual Leaky Bucket)

| GUI Parameter | Command Line Parameter |
|--|------------------------|
| Leaky Bucket Type (set to <i>Dual Leaky Bucket</i>) | DLB |
| Bucket Size | <Bucket Size> |
| Token Rate | <Token Rate> |
| Peak Rate | <Peak Rate> |
| Action on Packets Exceeding Thresholds | <Action> |

8.9.3.1.5 Configuring Fragmentation Parameters

To set fragmentation parameters, set **Enable Fragmentation** to Yes and set the dependent parameters listed in [Table 8-83](#); otherwise, set **Enable Fragmentation** to No.

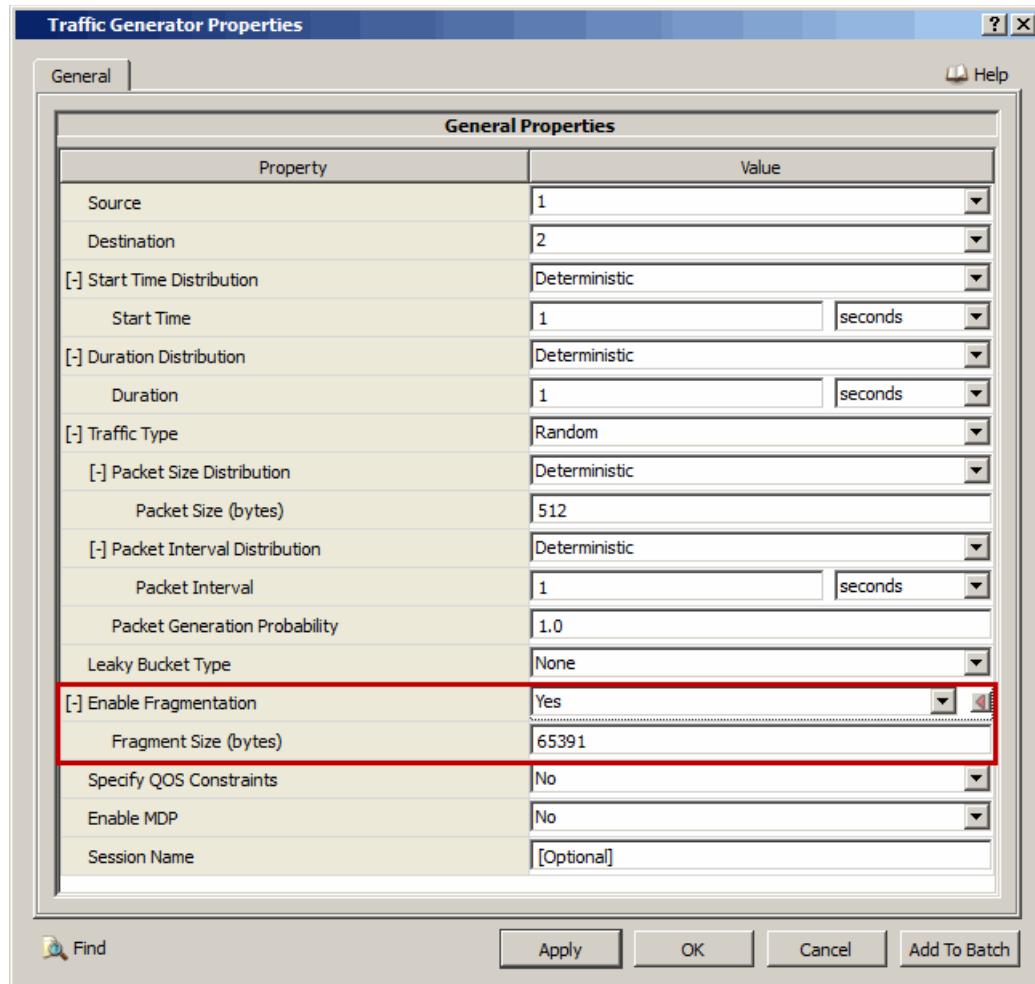


FIGURE 8-48. Setting Fragmentation Parameters

TABLE 8-83. Command Line Equivalent of Fragmentation Parameters

| GUI Parameter | Command Line Parameter |
|-----------------------------------|------------------------|
| Enable Fragmentation (set to Yes) | FRAGMENT-SIZE |
| Fragment Size | <Fragment-size> |

8.9.3.1.6 Configuring QoS Parameters

To configure the QoS parameters, perform the steps listed below.

1. Set **Specify QOS Constraints** to Yes and set the dependent parameters listed in [Table 8-84](#).

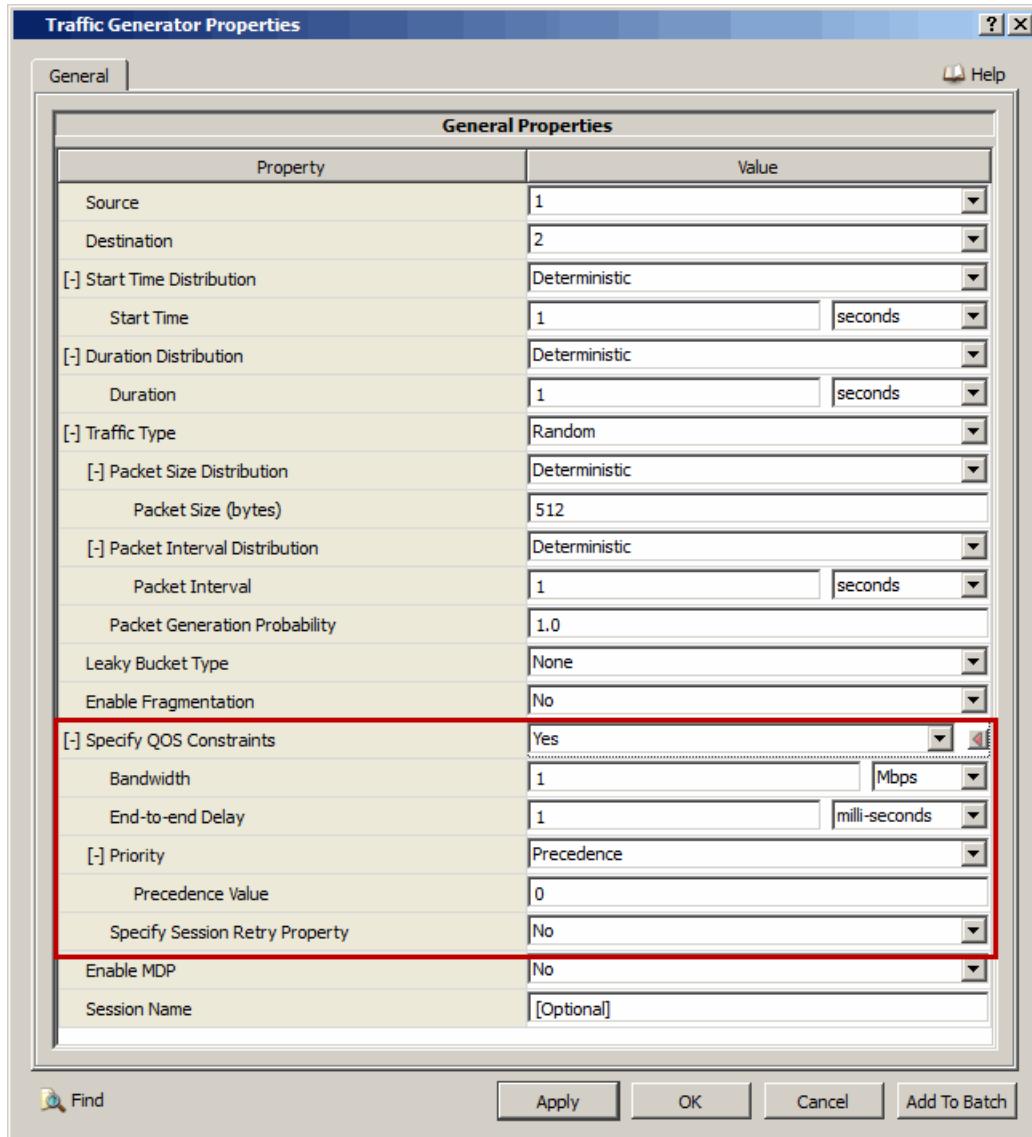


FIGURE 8-49. Setting QoS Parameters

TABLE 8-84. Command Line Equivalent of QoS Parameters

| GUI Parameter | Command Line Parameter |
|--------------------------------------|------------------------|
| Specify QOS Constraints (set to Yes) | CONSTRAINT |
| Bandwidth | <Bandwidth> |
| End-to-end Delay | <Delay> |

TABLE 8-84. Command Line Equivalent of QoS Parameters (Continued)

| GUI Parameter | Command Line Parameter |
|---|------------------------|
| Priority and dependent parameters | <Priority> |
| Specify Session Retry Property and dependent parameters | <Retry Property> |

Setting Parameters

- To set DSCP, Precedence, or TOS bits for packets, set **Priority** to *DSCP*, *Precedence*, or *TOS*, respectively.
 - To set session retry properties, set **Specify Session Retry Property** to *Yes*; otherwise, set **Specify Session Retry Property** to *No*.
2. If **Priority** is set to *DSCP*, *Precedence*, or *TOS*, set the dependent parameters listed in [Table 8-85](#), [Table 8-86](#), and [Table 8-87](#), respectively. [Figure 8-50](#) shows how to set the dependent parameters when **Priority** is set to *DSCP*. Setting dependent parameters for the other two options is similar.

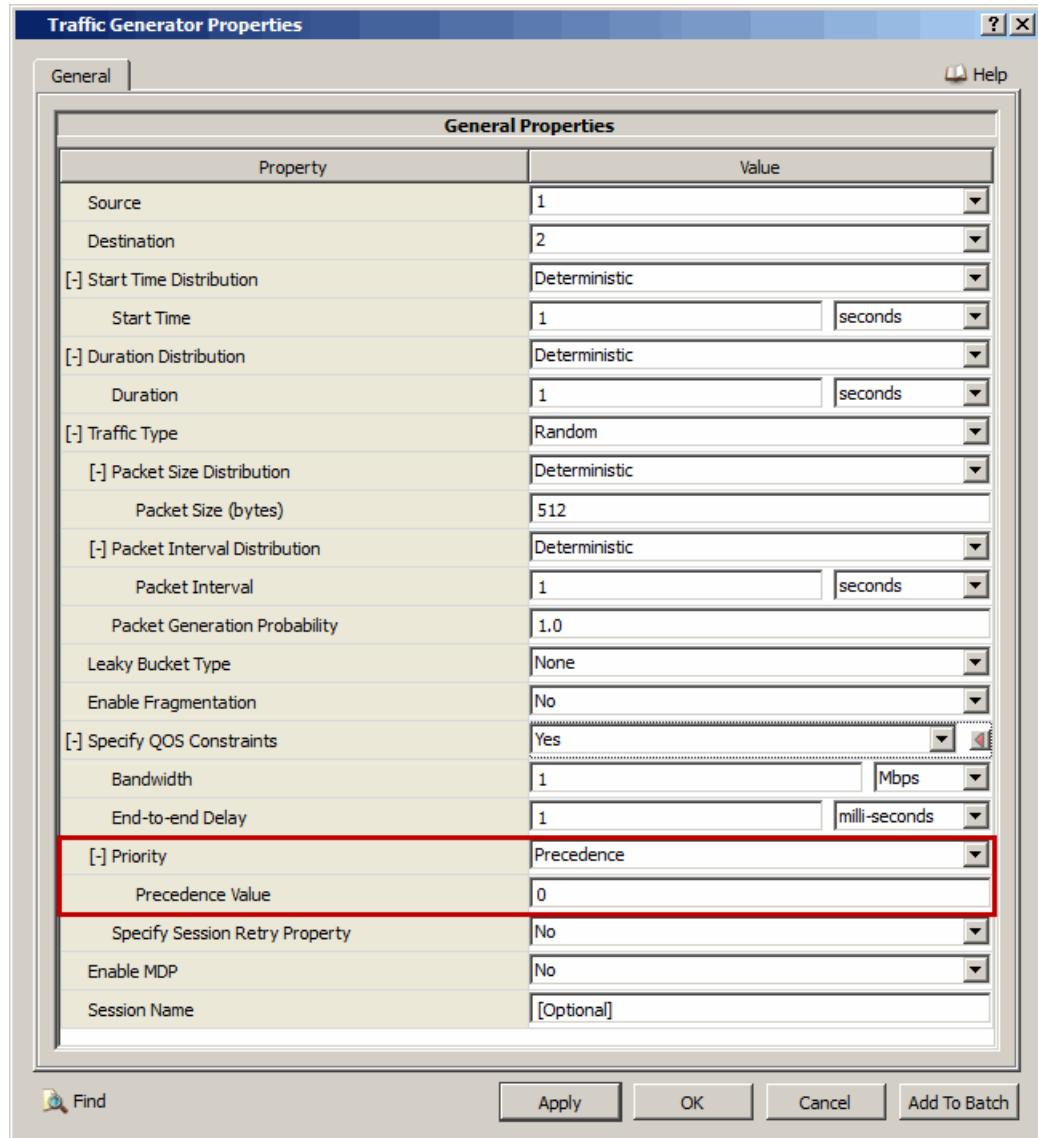


FIGURE 8-50. Setting DSCP Value

TABLE 8-85. Command Line Equivalent of Priority Parameters (Priority = DSCP)

| GUI Parameter | Command Line Parameter |
|------------------------|------------------------|
| Priority (set to DSCP) | DSCP |
| DSCP Value | <DSCP-value> |

TABLE 8-86. Command Line Equivalent of Priority Parameters (Priority = Precedence)

| GUI Parameter | Command Line Parameter |
|--------------------------------------|------------------------|
| Priority (set to <i>Precedence</i>) | PRECEDENCE |
| Precedence Value | <precedence-value> |

TABLE 8-87. Command Line Equivalent of Priority Parameters (Priority = TOS)

| GUI Parameter | Command Line Parameter |
|-------------------------------|------------------------|
| Priority (set to <i>TOS</i>) | TOS |
| TOS Value | <TOS-value> |

3. If Specify Session Retry Property to Yes, then set the dependent parameters listed in Table 8-88.

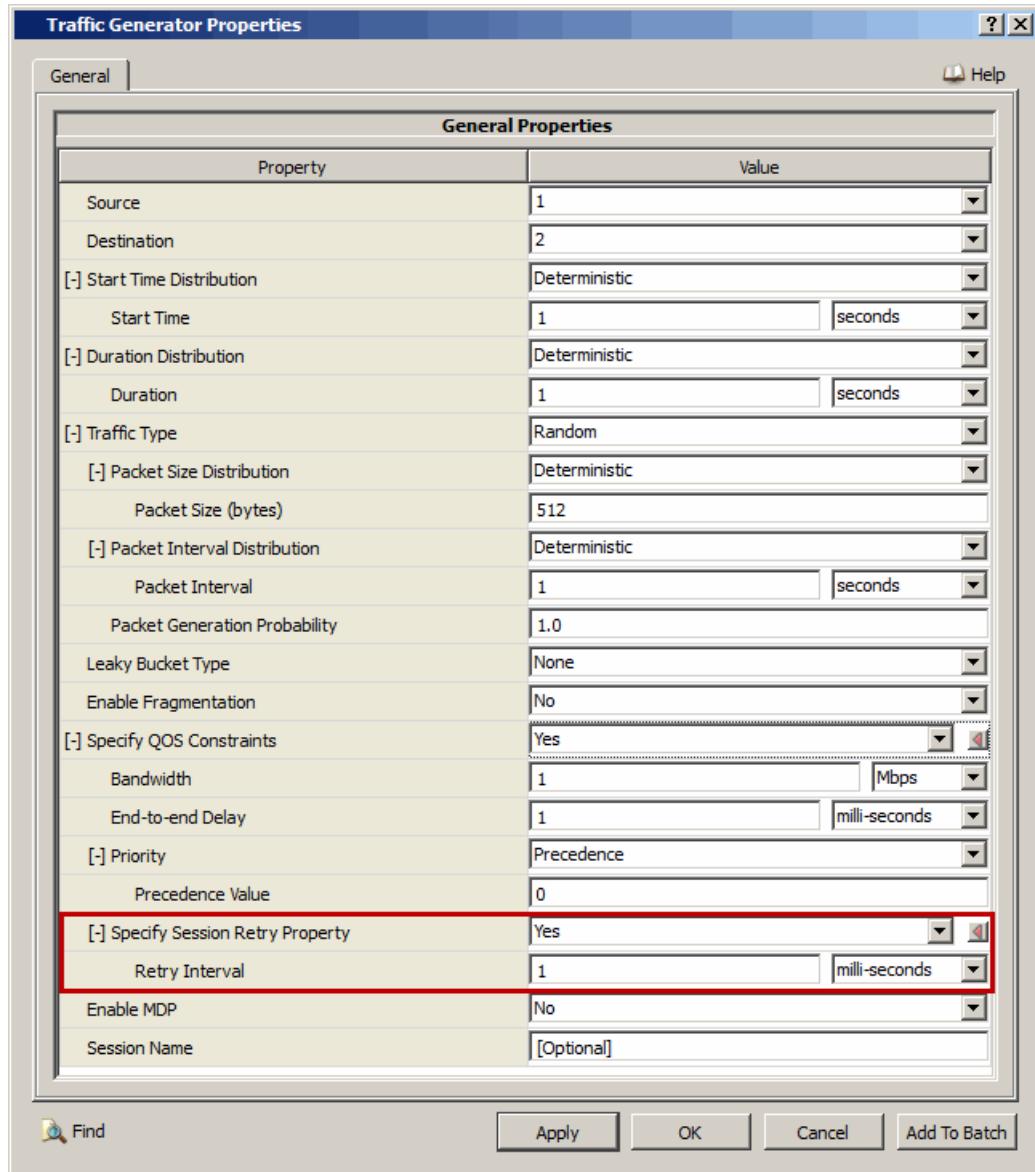


FIGURE 8-51. Setting Session Retry Parameters

TABLE 8-88. Command Line Equivalent of Session Retry Parameters

| GUI Parameter | Command Line Parameter |
|---|------------------------|
| Specify Session Retry Property (set to Yes) | RETRY-INTERVAL |
| Retry Interval | <interval> |

8.9.3.1.7 Configuring MDP Parameters

To configure the MDP parameters, perform the following steps:

1. To specify MDP parameters, set **Enable MDP** to Yes; otherwise, set **Enable MDP** to No.

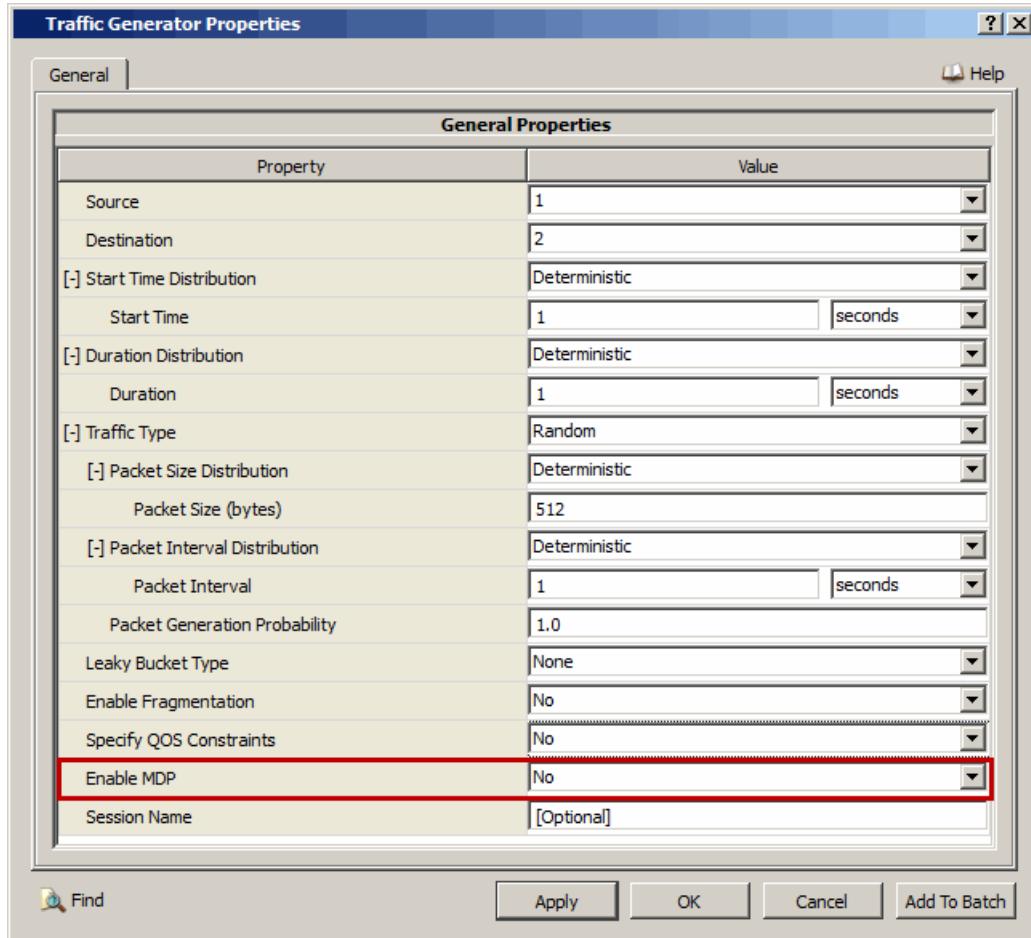


FIGURE 8-52. Enabling MDP

TABLE 8-89. Command Line Equivalent of MDP Parameters

| GUI Parameter | Command Line Parameter |
|-------------------------|------------------------|
| Enable MDP (set to Yes) | MDP - ENABLED |

2. If **Enable MDP** is set to Yes, then set the dependent parameters listed in [Table 8-90](#).

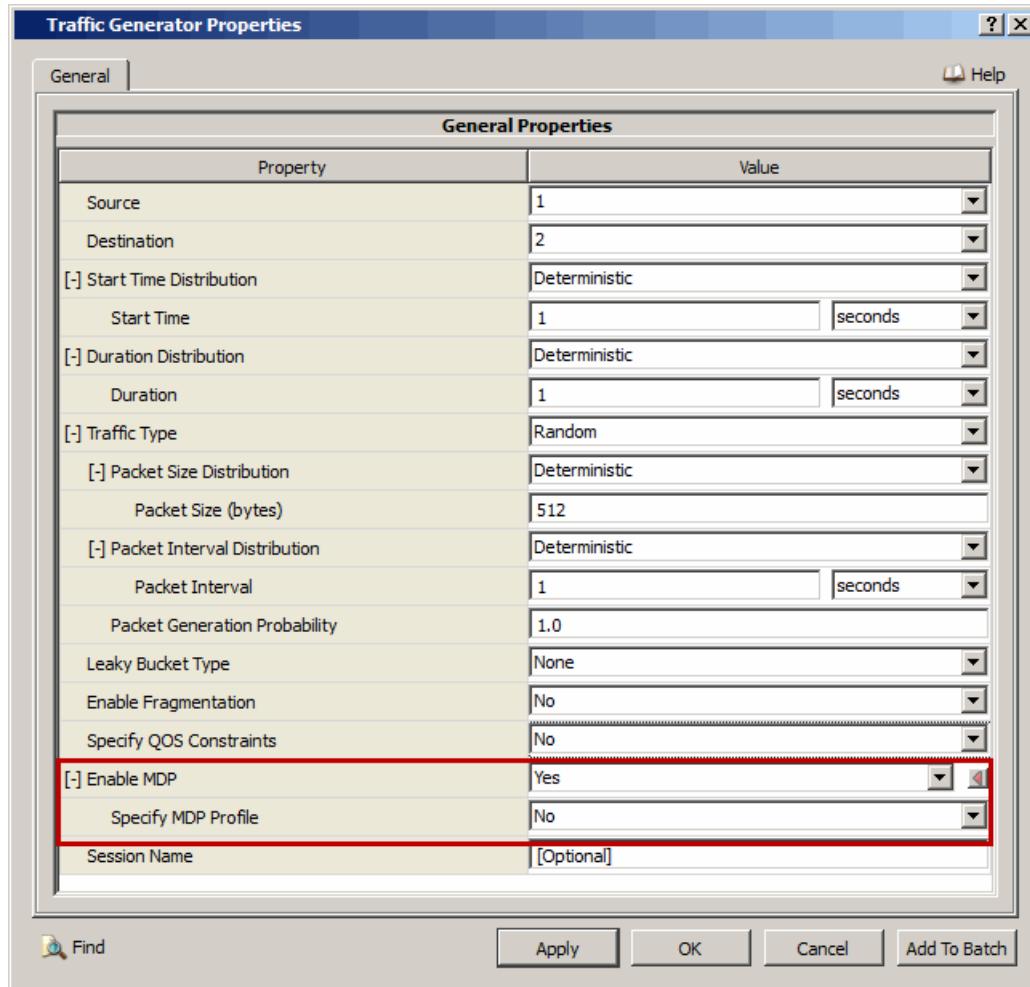


FIGURE 8-53. Configuring MDP Parameters

TABLE 8-90. Command Line Equivalent of MDP Configuration Parameters

| GUI Parameter | Command Line Parameter |
|----------------------------------|------------------------|
| Specify MDP Profile (set to Yes) | MDP - PROFILE |

Setting Parameters

- To specify an MDP profile, set **Specify MDP Profile** to Yes. To use the default MDP profile, set **Specify MDP Profile** to No.

3. If **Specify MDP Profile** is set to Yes, then set the parameters listed in [Table 8-91](#).

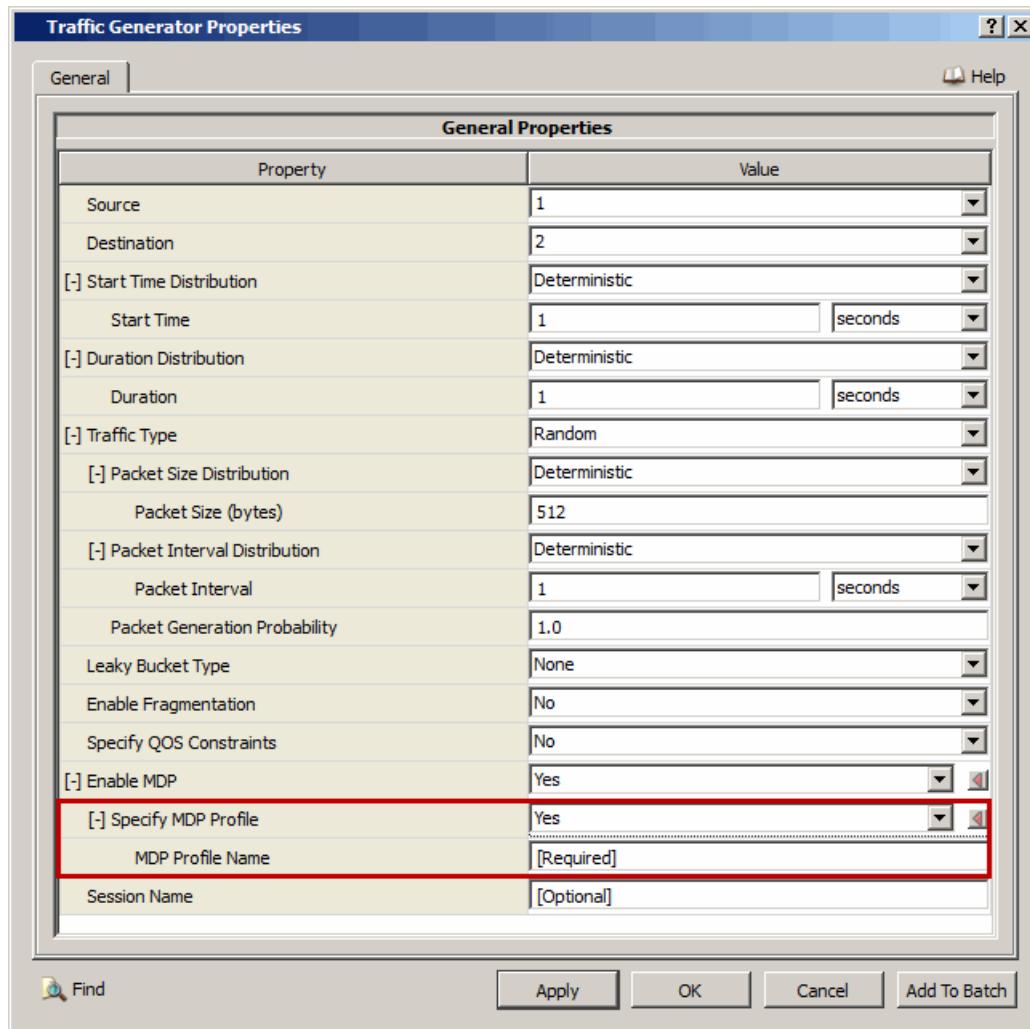


FIGURE 8-54. Specifying MDP Profile

TABLE 8-91. Command Line Equivalent of MDP Profile Parameters

| GUI Parameter | Command Line Parameter |
|------------------|------------------------|
| MDP Profile Name | <profile-name> |

8.9.3.2 Configuring a Single Host Traffic-Gen Session

To configure a single host Traffic-Gen session, perform the following steps:

1. Click the **TRAF GEN** button in the **Single Host Applications** tab of the Standard Toolset.
2. On the canvas, click on the client node.
3. Open the Traffic Generator Properties Editor by doing one of the following:
 - On the canvas, either double-click on the  symbol next to the client node or right-click on the  symbol and select **Properties** from the menu.
 - In the **Applications** tab of Table View, either double-click on the application row or right-click on the application row and select **Properties** from the menu.
4. Set the application parameters as described in [Section 8.9.3.2.1](#) to [Section 8.9.3.2.6](#).
5. Set the statistics parameters as described in [Section 8.9.3.3](#).

8.9.3.2.1 Configuring Source, Destination and Session Name, Parameters

Set the source, destination, and session name parameters listed in [Table 8-92](#).

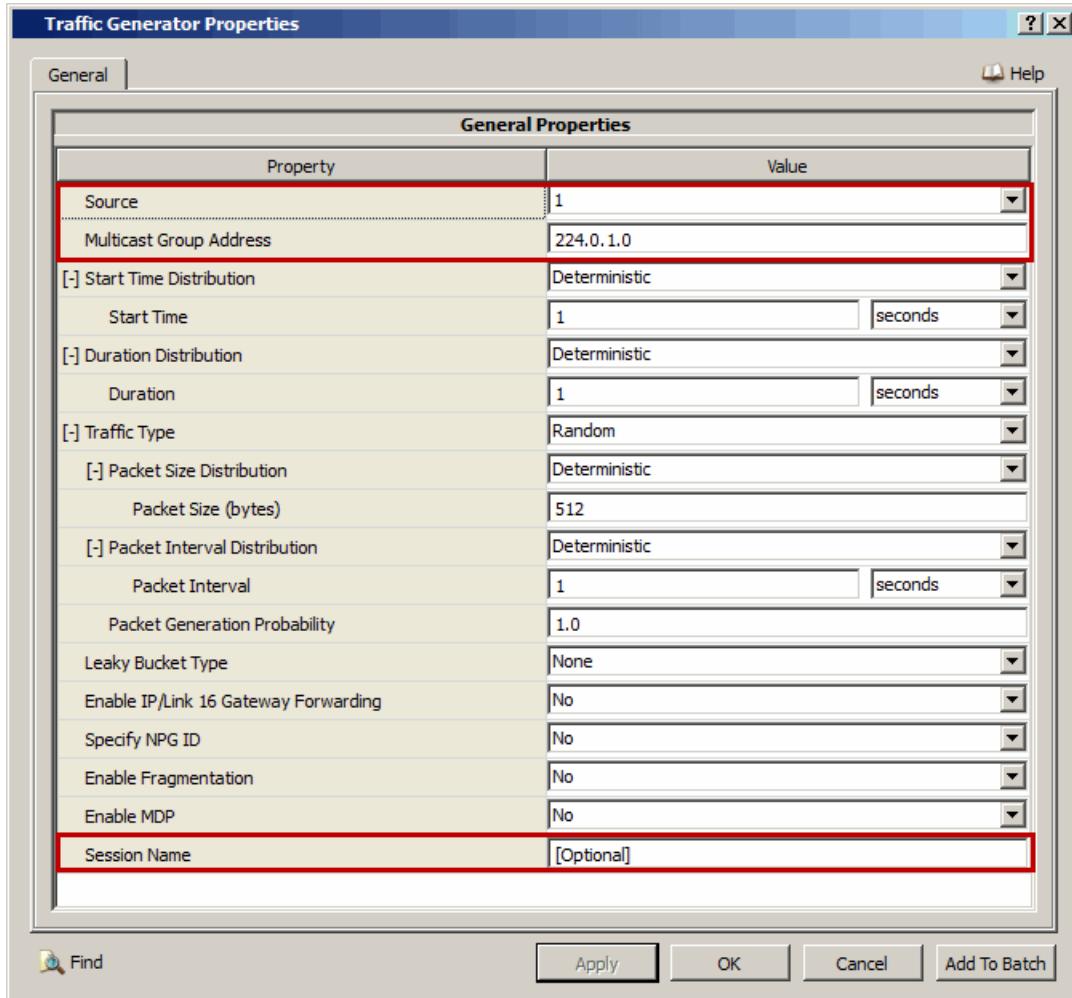


FIGURE 8-55. Setting Source and Destination Parameters

TABLE 8-92. Command Line Equivalent of Source and Destination Parameters

| GUI Parameter | Command Line Parameter |
|-------------------------|-------------------------------------|
| Source | <Source> |
| Multicast Group Address | <Destination> |
| Session Name | APPLICATION-NAME <application-name> |

Setting Parameters

- To specify an IP address as the source ID, set **Source** to one of the IP addresses listed in the drop-down list.
- Set **Multicast Group Address** to the address of the multicast group that is to receive traffic from the source.

8.9.3.2.2 Configuring Session Parameters

Configure the session parameters as described in [Section 8.9.3.1.2](#).

8.9.3.2.3 Configuring Traffic Parameters

Configure the traffic parameters as described in [Section 8.9.3.1.3](#).

8.9.3.2.4 Configuring Leaky Bucket Parameters

Configure the leaky bucket parameters as described in [Section 8.9.3.1.4](#).

8.9.3.2.5 Configuring Fragmentation Parameters

Configure the fragmentation parameters as described in [Section 8.9.3.1.5](#).

8.9.3.2.6 Configuring MDP Parameters

Configure the MDP parameters as described in [Section 8.9.3.1.7](#).

8.9.3.3 Configuring Statistics Parameters

Statistics for applications (including Traffic-Gen) can be collected at the global and node levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for Traffic-Gen, check the box labeled **Application** in the appropriate properties editor.

TABLE 8-93. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| Application | Global, Node | APPLICATION-STATISTICS |

8.9.4 Statistics

This section describes the file, database, and dynamic statistics of the Traffic-Gen model.

8.9.4.1 File Statistics

[Table 8-94](#) shows the Traffic-Gen statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 8-94. Traffic-Gen Statistics

| Statistics | Description |
|---------------------------------------|--|
| Traffic-Gen Client | |
| Server Address | Address of the server |
| Session Status | Current status of the session. |
| Total Data Units Dropped | Total number of data units dropped. |
| Unicast Session Start (seconds) | Time at which unicast session started, in seconds. |
| Unicast Session Finish (seconds) | Time at which unicast session ended, in seconds. |
| First Unicast Fragment Sent (seconds) | Time in second when first unicast fragment was sent. |
| Last Unicast Fragment Sent (seconds) | Time in second when last unicast fragment was sent. |

TABLE 8-94. Traffic-Gen Statistics (Continued)

| Statistics | Description |
|--|--|
| Total Unicast Fragments Sent (fragments) | Total number of unicast fragments sent. |
| First Unicast Message Sent (seconds) | Time in second when first unicast fragment was sent. |
| Last Unicast Message Sent (seconds) | Time in second when last unicast fragment was sent. |
| Total Unicast Messages Sent (messages) | Total number of unicast messages sent. |
| Total Unicast Data Sent (bytes) | Total number of data bytes sent. |
| Total Unicast Overhead Sent (bytes) | Total number of overhead bytes sent. |
| Unicast Offered Load (bits/second) | Unicast offered load. |
| Traffic-Gen Server | |
| Client Address | Address of the client. |
| Session Status | Current status of the session. |
| Unicast Session Start (seconds) | Time at which unicast session started, in seconds. |
| Unicast Session Finish (seconds) | Time at which unicast session ended, in seconds. |
| First Unicast Fragment Received (seconds) | Time in second when first unicast fragment was received. |
| Last Unicast Fragment Received (seconds) | Time in second when last unicast fragment was received. |
| Total Unicast Fragments Received (fragments) | Total number of unicast fragments received. |
| First Unicast Message Received (seconds) | Time in second when first unicast message was received. |
| Last Unicast Message Received (seconds) | Time in second when last unicast message was received. |
| Total Unicast Messages Received (messages) | Total number of unicast messages received. |
| Total Unicast Data Received (bytes) | Total number of unicast data bytes received. |
| Total Unicast Overhead Received (bytes) | Total number of unicast overhead bytes received. |
| Average Unicast End-to-End Delay (seconds) | Average unicast end-to-end delay. |
| Unicast Received Throughput (bits/second) | Unicast received throughput. |
| Average Unicast Jitter (seconds) | Average unicast jitter. |

8.9.4.2 Database Statistics

In addition to the file statistics, the Traffic-Gen model also enters statistics in various scenario statistics database tables. Refer to *QualNet Statistics Database User's Guide* for details.

8.9.4.3 Dynamic Statistics

No dynamic statistics are supported for the Traffic-Gen model.

8.9.5 Scenarios Included in QualNet

The QualNet distribution includes several sample scenarios for the Traffic-Gen model. All scenarios are located in the directory `QUALNET_HOME/scenarios/developer/traffic-gen`. [Table 8-95](#) lists the sub-directory where each scenario is located.

TABLE 8-95. Traffic-Gen Scenarios Included in QualNet

| Scenario | Description |
|----------|--|
| exp | Shows the Traffic-Gen application with data length of traffic property is set as EXP distribution type. |
| tpd | Shows the Traffic-Gen application with data length of traffic property is set as TPD distribution type. |
| tpd4 | Shows the Traffic-Gen application with data length of traffic property is set as TPD4 distribution type. |

8.10 Trace File-based Traffic Generator (Traffic-Trace)

8.10.1 Description

Traffic-Trace simulates a trace file-based traffic generator. Trace files must have two columns; the first for time interval between data items, and the second for the length of each data item.

8.10.2 Command Line Configuration

Configuration Requirements

In order to configure a Traffic-Trace session with QoS constraints, a path must exist from the source to each destination node along which QOSPF (Quality of Service Extensions to OSPF) is enabled. See *Multimedia and Enterprise Model Library* for details of configuring QOSPF.

Application Configuration File Parameters

To specify Traffic-Trace traffic, include the following statement in the application configuration (.app) file:

```
TRAFFIC-GEN <Source> <Destination> <Session Parameters>
    <Traffic Parameters> <Leaky Bucket Parameters>
    [<QoS Parameters>] [<MDP Parameters>]
```

Note: All parameters should be entered on the same line.

The Traffic-Trace parameters are described in [Table 8-96](#).

TABLE 8-96. Traffic-Trace Parameters

| Element | Description | | | | | | | | |
|---------------------------|--|---------------|---|--------------|---|-------------|-------------------------------|----------|---|
| <Source> | Node ID or IP address of the source node. | | | | | | | | |
| <Destination> | Node ID or IP address of the destination node or a multicast address. | | | | | | | | |
| <Session Parameters> | <p>Session properties (start time and duration) of the connection. The session properties are specified in the following format:</p> <pre><Start Time> <Duration></pre> <p>where</p> <table> <tr> <td><Start Time></td> <td>Time when the session starts. This is specified as a time distribution (see note 1).</td> </tr> <tr> <td><Duration></td> <td>Length of the session. This is specified as a time distribution (see note 1).</td> </tr> </table> | <Start Time> | Time when the session starts. This is specified as a time distribution (see note 1). | <Duration> | Length of the session. This is specified as a time distribution (see note 1). | | | | |
| <Start Time> | Time when the session starts. This is specified as a time distribution (see note 1). | | | | | | | | |
| <Duration> | Length of the session. This is specified as a time distribution (see note 1). | | | | | | | | |
| <Traffic Parameters> | <p>Traffic properties (size of packets and inter-packet times) of the connection. For Traffic-Trace, these are read from a file. The traffic properties are specified in the following format:</p> <pre>TRC <Trace-file></pre> <p>where</p> <table> <tr> <td><Trace-file></td> <td>Name of the file that contains the traffic trace. The format of the Traffic Trace file is described in Section 8.10.2.1.</td> </tr> </table> | <Trace-file> | Name of the file that contains the traffic trace. The format of the Traffic Trace file is described in Section 8.10.2.1 . | | | | | | |
| <Trace-file> | Name of the file that contains the traffic trace. The format of the Traffic Trace file is described in Section 8.10.2.1 . | | | | | | | | |
| <Leaky Bucket Parameters> | <p>Leaky bucket properties for traffic shaping. There are three options of specifying leaky bucket properties.</p> <p>If traffic shaping is not used, use the following format:</p> <pre>NLB</pre> <p>If leaky bucket is used, for traffic shaping use the following format:</p> <pre>LB <Bucket Size> <Token Rate> <Action></pre> <p>If dual leaky bucket is used for traffic shaping, use the following format:</p> <pre>DLB <Bucket Size> <Token Rate> <Peak Rate> <Action></pre> <p>where</p> <table> <tr> <td><Bucket Size></td> <td>Bucket size, in bytes.</td> </tr> <tr> <td><Token Rate></td> <td>Token generation rate, in bps.</td> </tr> <tr> <td><Peak Rate></td> <td>Maximum allowed rate, in bps.</td> </tr> <tr> <td><Action></td> <td>Action to perform. This can be DROP or DELAY.</td> </tr> </table> | <Bucket Size> | Bucket size, in bytes. | <Token Rate> | Token generation rate, in bps. | <Peak Rate> | Maximum allowed rate, in bps. | <Action> | Action to perform. This can be DROP or DELAY. |
| <Bucket Size> | Bucket size, in bytes. | | | | | | | | |
| <Token Rate> | Token generation rate, in bps. | | | | | | | | |
| <Peak Rate> | Maximum allowed rate, in bps. | | | | | | | | |
| <Action> | Action to perform. This can be DROP or DELAY. | | | | | | | | |

TABLE 8-96. Traffic-Trace Parameters (Continued)

| Element | Description | | | | | | | | | | | | | | | | |
|------------------|--|----------------|--|---------|--|------------|---|--|--|------------------|--|--|---------------------------|--|-------|--|---|
| <QoS Parameters> | <p>QoS properties of the connection.</p> <p>Note: QoS properties specification is optional can only be included for client-server sessions.</p> <p>The QoS properties are specified in the following format:</p> <pre>CONSTRAINT <Bandwidth> <Delay> [<Priority>] [<Retry Property>]</pre> <p>where</p> <table> <tr> <td data-bbox="589 620 768 650"><Bandwidth></td> <td data-bbox="882 620 1163 650">QoS Bandwidth, in bps.</td> </tr> <tr> <td data-bbox="589 663 703 692"><Delay></td> <td data-bbox="882 663 1290 692">QoS end-to-end delay, in seconds.</td> </tr> <tr> <td data-bbox="589 705 752 734"><Priority></td> <td data-bbox="882 705 1339 734">Priority given to a session. See note 2.</td> </tr> <tr> <td data-bbox="882 747 899 777"></td> <td data-bbox="882 747 1323 777">Note: Priority specification is optional.</td> </tr> <tr> <td data-bbox="589 789 850 819"><Retry Property></td> <td data-bbox="882 789 1405 853">Retry characteristics, which is specified in the following format:</td> </tr> <tr> <td data-bbox="931 865 1144 895"></td> <td data-bbox="931 865 1323 895">RETRY-INTERVAL <interval></td> </tr> <tr> <td data-bbox="882 908 948 937"></td> <td data-bbox="882 908 948 937">where</td> </tr> <tr> <td data-bbox="931 950 980 979"></td> <td data-bbox="931 950 1388 1009"><interval> Retry interval, specified as a time value.</td> </tr> </table> <p>Note: Retry property specification is optional.</p> | <Bandwidth> | QoS Bandwidth, in bps. | <Delay> | QoS end-to-end delay, in seconds. | <Priority> | Priority given to a session. See note 2. | | Note: Priority specification is optional. | <Retry Property> | Retry characteristics, which is specified in the following format: | | RETRY-INTERVAL <interval> | | where | | <interval> Retry interval, specified as a time value. |
| <Bandwidth> | QoS Bandwidth, in bps. | | | | | | | | | | | | | | | | |
| <Delay> | QoS end-to-end delay, in seconds. | | | | | | | | | | | | | | | | |
| <Priority> | Priority given to a session. See note 2. | | | | | | | | | | | | | | | | |
| | Note: Priority specification is optional. | | | | | | | | | | | | | | | | |
| <Retry Property> | Retry characteristics, which is specified in the following format: | | | | | | | | | | | | | | | | |
| | RETRY-INTERVAL <interval> | | | | | | | | | | | | | | | | |
| | where | | | | | | | | | | | | | | | | |
| | <interval> Retry interval, specified as a time value. | | | | | | | | | | | | | | | | |
| <MDP Parameters> | <p>MDP parameters of the connection.</p> <p>Note: MDP parameter specification is optional. If MDP parameters are not specified, the application does not run with MDP.</p> <p>The MDP parameters are specified in the following format:</p> <pre>MDP-ENABLED [MDP-PROFILE <profile-name>]</pre> <p>where</p> <table> <tr> <td data-bbox="589 1360 817 1389"><profile-name></td> <td data-bbox="882 1360 1388 1419">Name of the MDP profile to be used with the application.</td> </tr> <tr> <td data-bbox="882 1431 899 1461"></td> <td data-bbox="882 1431 1356 1529">This should be the name of a MDP profile defined in the MDP profile file (see Section 7.2.4.3).</td> </tr> <tr> <td data-bbox="882 1541 899 1571"></td> <td data-bbox="882 1541 1388 1634">This MDP profile takes precedence over the MDP profile associated with the node in the scenario configuration file, if any.</td> </tr> </table> <p>Note: If parameter MDP-ENABLED is included but parameter MDP-PROFILE is not included, and a MDP profile is specified for the node in the scenario configuration file, then the MDP profile specified for the node is used.</p> <p>If parameter MDP-ENABLED is included but parameter MDP-PROFILE is not included, and a MDP profile is not specified for the node in the scenario configuration file, then default MDP values for the MDP profile parameters are used (see Section 7.2.4.3).</p> | <profile-name> | Name of the MDP profile to be used with the application. | | This should be the name of a MDP profile defined in the MDP profile file (see Section 7.2.4.3). | | This MDP profile takes precedence over the MDP profile associated with the node in the scenario configuration file, if any. | | | | | | | | | | |
| <profile-name> | Name of the MDP profile to be used with the application. | | | | | | | | | | | | | | | | |
| | This should be the name of a MDP profile defined in the MDP profile file (see Section 7.2.4.3). | | | | | | | | | | | | | | | | |
| | This MDP profile takes precedence over the MDP profile associated with the node in the scenario configuration file, if any. | | | | | | | | | | | | | | | | |

Notes: 1. Integer and Time Distributions: Several Traffic-Trace parameters are specified as random number distributions. Five random number distributions are supported: deterministic, uniform, exponential, truncated Pareto, and 4-parameter truncated Pareto.

- The deterministic distribution is specified as:

```
DET <det-val>
```

It always returns `<det-val>` as the value.

- The uniform distribution is specified as:

```
UNI <uni-val-1> <uni-val-2>
```

It returns a value uniformly distributed between `<uni-val-1>` and `<uni-val-2>`.

- The exponential distribution is specified as:

```
EXP <exp-val>
```

It returns a value from an exponential distribution with `<exp-val>` as the mean.

- The truncated Pareto distribution is specified as:

```
TPD <tp-val-1> <tp-val-2> <tp-alpha>
```

It returns a value from a truncated Pareto distribution with `<tp-val-1>` as the lower end of the range, `<tp-val-2>` as the upper limit of the truncation, and `<tp-alpha>` as the shape parameter.

- The 4-parameter truncated Pareto distribution is specified as:

```
TPD4 <tp4-val-1> <tp-val-2> <tp4-val-3> <tp4-alpha>
```

It returns a value from a truncated Pareto distribution with `<tp4-val-1>` as the lower end of the range, `<tp4-val-2>` as the lower limit of the truncation, `<tp4-val-3>` as the upper limit of the truncation, and `<tp4-alpha>` as the shape parameter.

For integer distributions, `<det-val>`, `<uni-val-1>`, `<uni-val-2>`, `<exp-val>`, `<tp-val-1>`, `<tp-val-2>`, `<tp-alpha>`, `<tp4-val-1>`, `<tp4-val-2>`, `<tp4-val-3>`, and `<tp4-alpha>` are integer values, e.g., 0, 10, 15, etc.

For time distributions, `<det-val>`, `<uni-val-1>`, `<uni-val-2>`, `<exp-val>`, `<tp-val-1>`, `<tp-val-2>`, `<tp-alpha>`, `<tp4-val-1>`, `<tp4-val-2>`, `<tp4-val-3>`, and `<tp4-alpha>` are time values, e.g., 5S, 0.5MS, 100US, etc.

2. Priority Specification: Priority can be specified by including a TOS specification, DSCP specification, or Precedence specification.

- TOS specification has the following format:

```
TOS <TOS-value>
```

where `<TOS-value>` is the value of the TOS bits of the IP header. `<TOS-value>` should be an integer in the range [0, 255].

- DSCP specification has the following format:

```
DSCP <DSCP-value>
```

where `<DSCP-value>` is the value of the DSCP bits of the IP header. `<DSCP-value>` should be an integer in the range [0, 63].

- Precedence specification has the following format:

```
PRECEDENCE <precedence-value>
```

where `<precedence-value>` is the value of the Precedence bits of the IP header. `<precedence-value>` should be an integer in the range [0, 7].

At most one of the three parameters TOS, DSCP, or PRECEDENCE can be specified. If a TOS specification, DSCP specification, or Precedence specification is not included, PRECEDENCE 0 is used as default.

Scenario Configuration File Parameters

Table 8-97 describes the Traffic-Trace parameters that can be specified in the scenario configuration (.config) file.

TABLE 8-97. Traffic-Trace Scenario Configuration File Parameters

| Parameter | Value | Description |
|--|--|--|
| APPLICATION-STATISTICS <i>Optional</i> Scope: Global, Node | List: • YES • NO <i>Default:</i> NO | Indicates whether statistics collection is enabled for applications (including Traffic-Trace). |

Examples of Parameter Usage

The following are examples of Traffic-Trace configuration:

1. This is an example of video traffic that is not shaped with a leaky bucket. The start time and duration are 0 seconds and 600 seconds, respectively.

```
TRAFFIC-TRACE 1 2 DET 0 DET 600 TRC soccer.trc NOLB
```

The video traffic is characterized with an average rate of about 0.64 Mbps and a peak rate of a little more than 4 Mbps. Figure 8-56 shows the rate of the traffic and also its overall average. The average

rate looks right, but the peak is much less than 4 Mbps. The reason is that the rate graph was generated with a time sliding window of size 1 second for which outgoing traffic rate is averaged. Thus, any instant peak was not clearly captured.

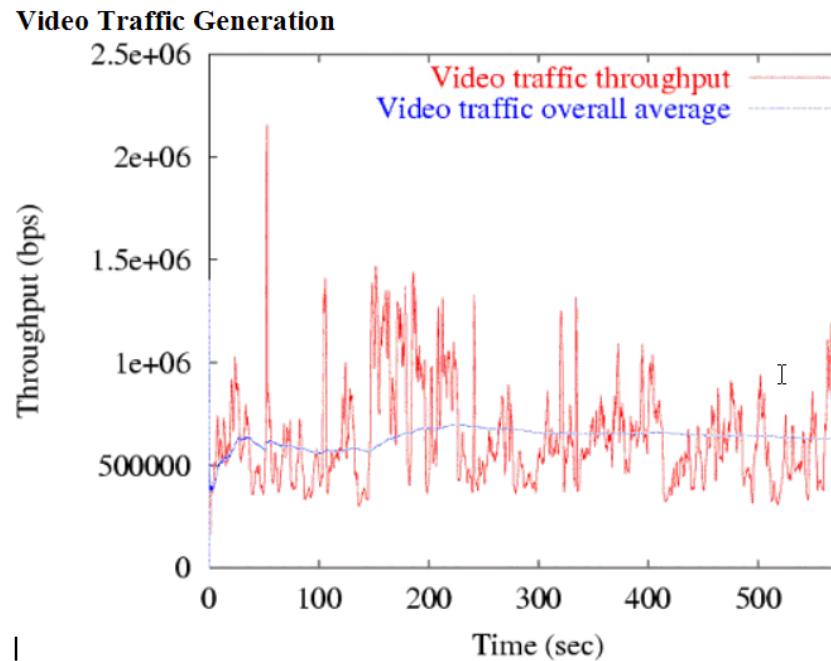
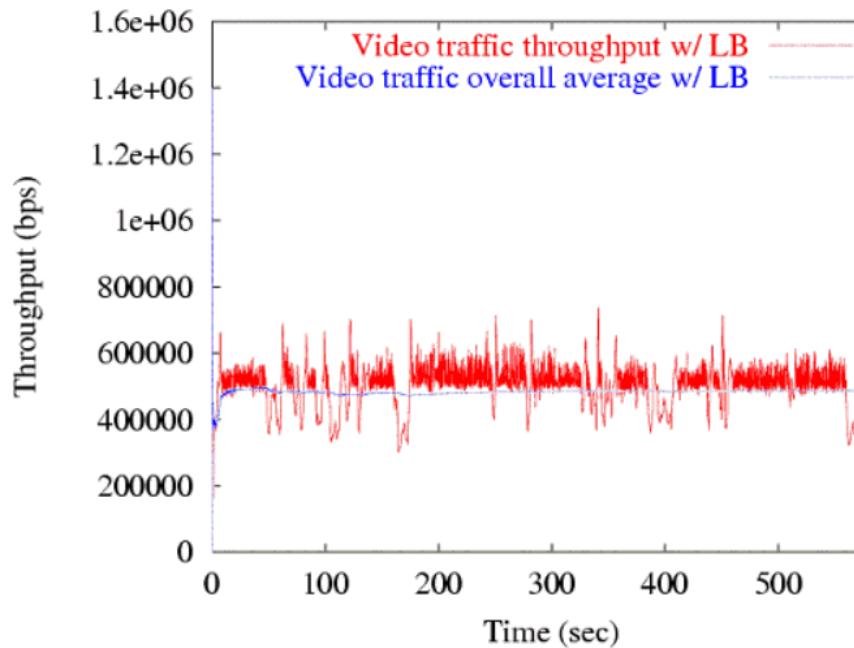


FIGURE 8-56. Video Traffic Generation

2. This is an example of video traffic that uses a leaky bucket to regulate the traffic. Since leaky bucket regulates only burst size and average rate, choose an arbitrarily smaller value for the average rate than the actual average rate of the video trace. This example is configured to delay packets rather than drop them.

```
TRAFFIC-TRACE 1 2 DET 0 DET 600 TRC soccer.trc LB 30000 500000 DELAY
```

In [Figure 8-57](#), leaky bucket parameters, the bucket size in bytes and the average in bps, are specified as 30 KB and 500 Kbps, respectively. The reason for 30-KB bucket size is that since the trace file contains the biggest size data of a little less than 30 KB, the bucket size should be at least larger than that. The reason for 500-Kbps average rate is to show that the overall traffic average is suppressed by this parameter. This suppressed overall average is shown in [Figure 8-57](#).

Video Traffic with Leaky Bucket**FIGURE 8-57. Video Traffic with Leaky Bucket**

3. This is an example of video traffic that uses a dual leaky bucket to regulate the traffic. Like regular leaky bucket, dual leaky bucket controls the burst size and the overall average rate. In addition, dual leaky bucket also controls the peak rate. The parameters for the dual leaky bucket are the bucket size of 30 KB, the average rate of 800 Kbps (larger than the actual average rate of the trace), and the peak rate of 1 Mbps (much smaller than the actual peak rate of the trace). Consequently, the peak rate of the traffic is regulated by the dual leaky bucket and this is clearly shown in [Figure 8-58](#).

```
TRAFFIC-TRACE 1 2 DET 0 DET 600 TRC soccer.trc DLB 30000 800000 1000000  
DELAY
```

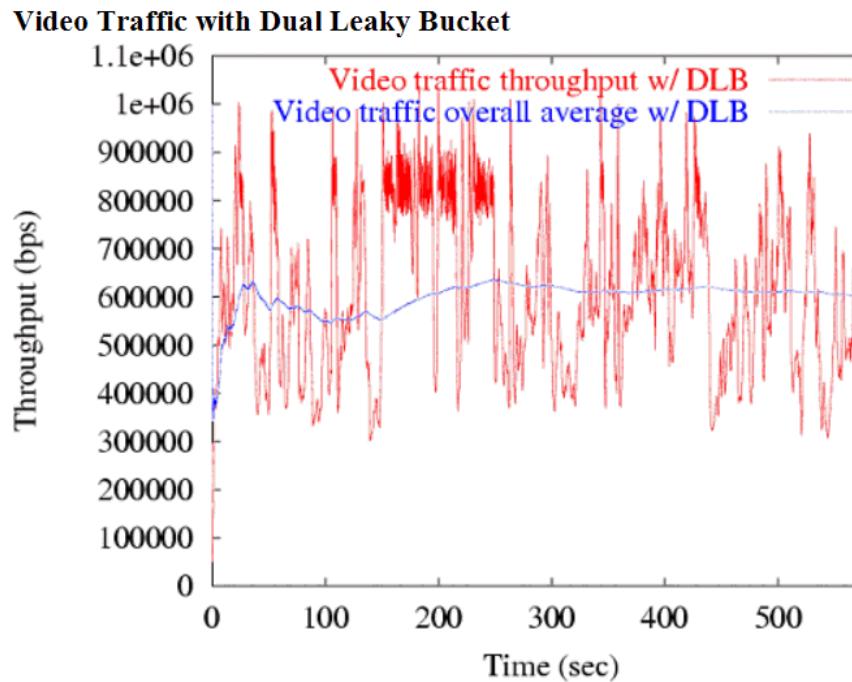


FIGURE 8-58. Video Traffic with Dual Leaky Bucket

4. This is the same as the previous example, except that Traffic-Trace runs with MDP and uses the MDP profile associated with node 1. If there is no MDP profile associated with node 1, default values for MDP profile parameters are used.

```
TRAFFIC-TRACE 1 2 DET 0 DET 600 TRC soccer.trc DLB 30000 800000 1000000
DELAY MDP-ENABLED
```

5. This is the same as the previous example, except that MDP uses the user-defined MDP profile profile-1.

```
TRAFFIC-TRACE 1 2 DET 0 DET 600 TRC soccer.trc DLB 30000 800000 1000000
DELAY MDP-ENABLED MDP-PROFILE profile-1
```

8.10.2.1 Format of the Traffic Trace File

The Traffic Trace file lists the size of packets and the inter-packet times.

Each line in the Traffic Trace file has the following format:

```
<next-packet-interval> <packet-size>
```

where

| | |
|---|--|
| <code><next-packet-interval></code> | Time, in QualNet time format, after which the next packet is sent. (The size of the next packet is <code><packet-size></code> on the next line.) |
|---|--|

Note: The first packet is sent when the session starts (as specified by the parameter `<start-time>`). Each of the other packets is sent at (`<next-packet-interval>` of the previous packet + time the previous packet was sent).

| | |
|----------------------------------|-------------------------------|
| <code><packet-size></code> | Size of the packet, in bytes. |
|----------------------------------|-------------------------------|

Example

Consider the following trace file:

```
10 200
15 150
10 300
15 100
```

Packets are sent at the following times:

- The first packet is 200 bytes long and is sent at `<start-time>`.
- The second packet is 150 bytes long and is sent 10 seconds after the first packet (at `<start-time>` + 10 seconds).
- The third packet is 300 bytes long and is sent 15 seconds after the second packet (at `<start-time>` + 25 seconds), and so on.

8.10.3 GUI Configuration

[Section 8.10.3.1](#) describes how to configure a Traffic-Trace session between two nodes. [Section 8.10.3.2](#) describes how to configure a single host Traffic-Trace session. [Section 8.10.3.3](#) describes how to configure statistics parameters for applications (including Traffic-Trace).

Configuration Requirements

In order to configure a Traffic-Trace session with QoS constraints, a path must exist from the source to each destination node along which QOSPF (Quality of Service Extensions to OSPF) is enabled. See *Multimedia and Enterprise Model Library* for details of configuring QOSPF.

8.10.3.1 Configuring a Client-Server Traffic-Trace Session

To configure a Traffic-Trace session between two nodes, perform the following steps:

1. Click the **TRAF TRC** button in the **Applications** tab of the Standard Toolset.
2. On the canvas, click on the source node, drag the mouse to the destination node, and release.

3. Open the Traffic Trace Properties Editor by doing one of the following:
 - Right-click in the application link on the canvas and select **Properties** from the menu.
 - In the **Applications** tab of Table View either double-click on the application row or right-click on the application row and select **Properties** from the menu.
4. Set the application parameters as described in [Section 8.10.3.1.1](#) to [Section 8.10.3.1.6](#).
5. Set the statistics parameters as described in [Section 8.10.3.3](#).

8.10.3.1.1 Configuring Source and Destination Parameters

Set the source and destination parameters listed in [Table 8-98](#).

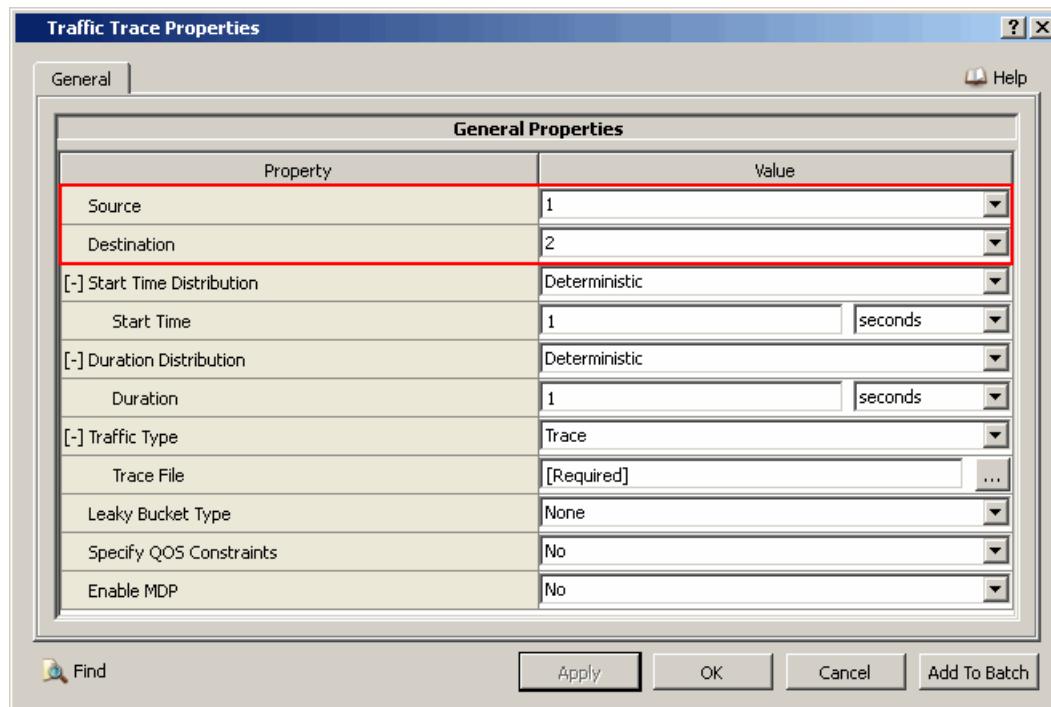


FIGURE 8-59. Setting Source and Destination Parameters

TABLE 8-98. Command Line Equivalent of Source and Destination Parameters

| GUI Parameter | Command Line Parameter |
|---------------|------------------------|
| Source | <Source> |
| Destination | <Destination> |

Setting Parameters

- To specify an IP address as the source (destination) ID, set **Source (Destination)** to one of the IP addresses listed in the drop-down list.

8.10.3.1.2 Configuring Session Parameters

To configure the session parameters, perform the following steps:

1. Set the parameters **Start Time Distribution** and **Duration Distribution**.

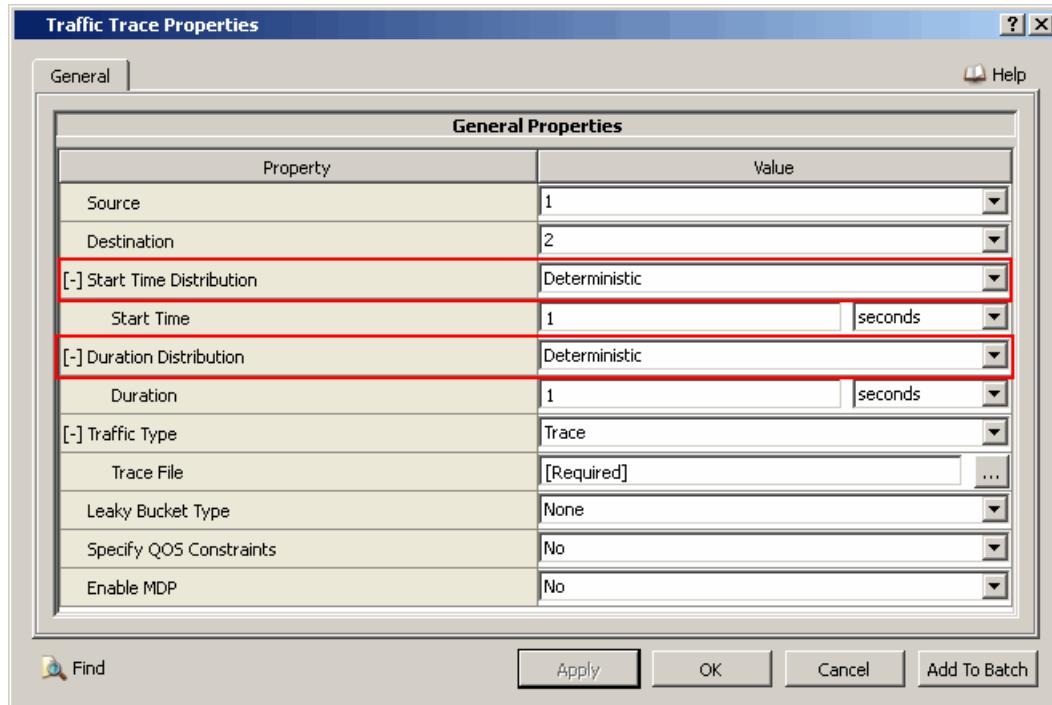


FIGURE 8-60. Setting Session Parameters

TABLE 8-99. Command Line Equivalent of Session Parameters

| GUI Parameter | Command Line Parameter |
|--|------------------------|
| Start Time Distribution and dependent parameters | <Start Time> |
| Duration Distribution and dependent parameters | <Duration> |

2. For each of the parameters **Start Time Distribution** and **Duration Distribution**, set the dependent parameters of the selected distribution as described in [Section 8.9.3.1.2](#).

8.10.3.1.3 Configuring Traffic Parameters

To configure the traffic parameters, set **Trace File** to the name of the Traffic Trace file. See [Section 8.10.2.1](#) for the format of the Traffic Trace file.

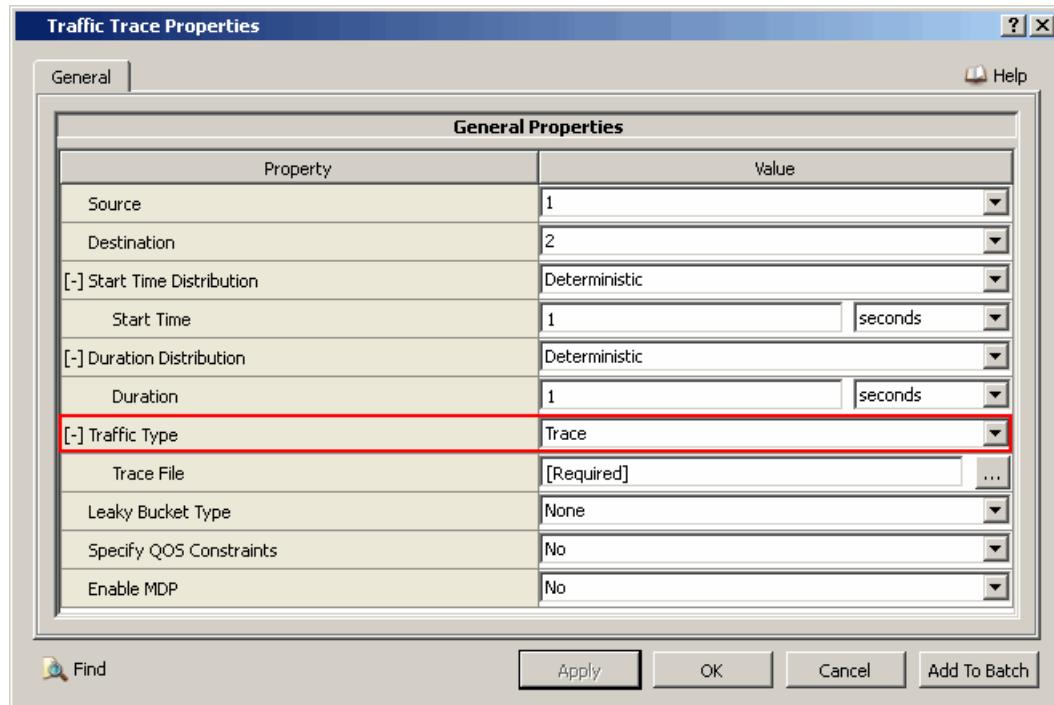


FIGURE 8-61. Setting Traffic Parameters

TABLE 8-100. Command Line Equivalent of Traffic Parameters

| GUI Parameter | Command Line Parameter |
|-------------------------------------|------------------------|
| Traffic Type (set to <i>Trace</i>) | TRC |
| Trace File | <Trace-file> |

Setting Parameters

- Set **Trace File** to the name of the Traffic Trace file. See [Section 8.10.2.1](#) for the format of the Traffic Trace file.

8.10.3.1.4 Configuring Leaky Bucket Parameters

To configure the leaky bucket parameters, perform the following steps:

1. Set the parameter **Leaky Bucket Type**.

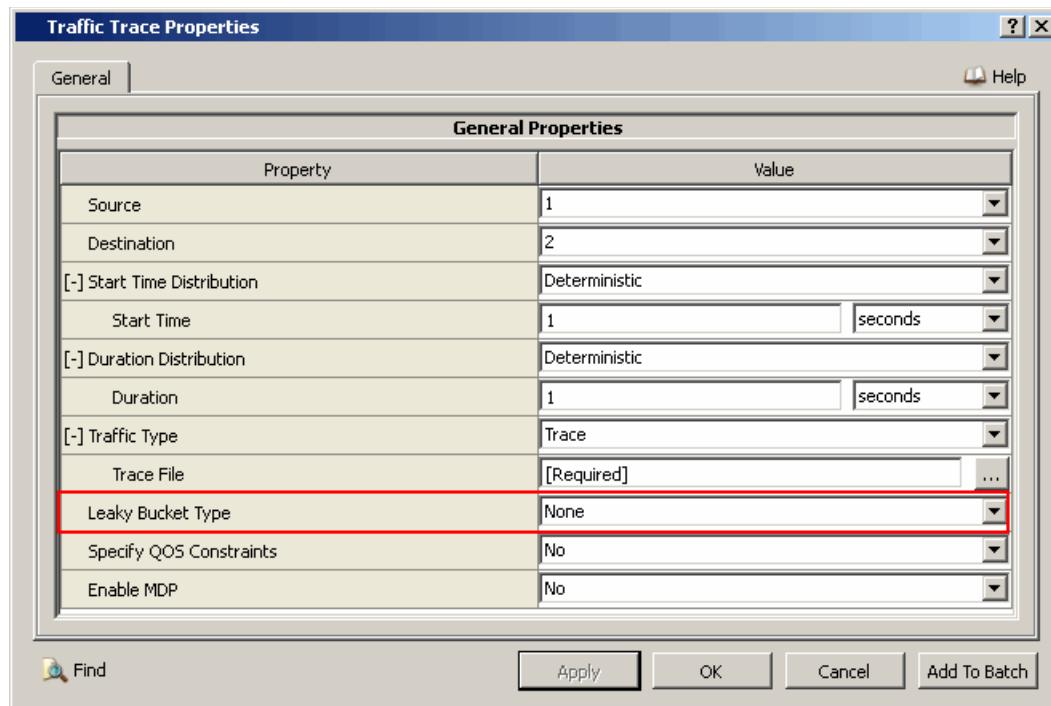


FIGURE 8-62. Configuring Leaky Bucket Parameters

TABLE 8-101. Command Line Equivalent of Leaky Bucket Parameters

| GUI Parameter | Command Line Parameter |
|--|------------------------------|
| Leaky Bucket Type and dependent parameters | <Leaky Bucket Specification> |

Setting Parameters

- To use leaky bucket for traffic shaping, set **Leaky Bucket** to *Leaky Bucket*. To use dual leaky bucket for traffic shaping, set **Leaky Bucket** to *Dual Leaky Bucket*.
2. Set the dependent parameters of **Leaky Bucket** as described in [Section 8.9.3.1.4](#).

8.10.3.1.5 Configuring QoS Parameters

To configure the QoS parameters, perform the steps listed below.

1. Set **Specify QOS Constraints** to Yes.

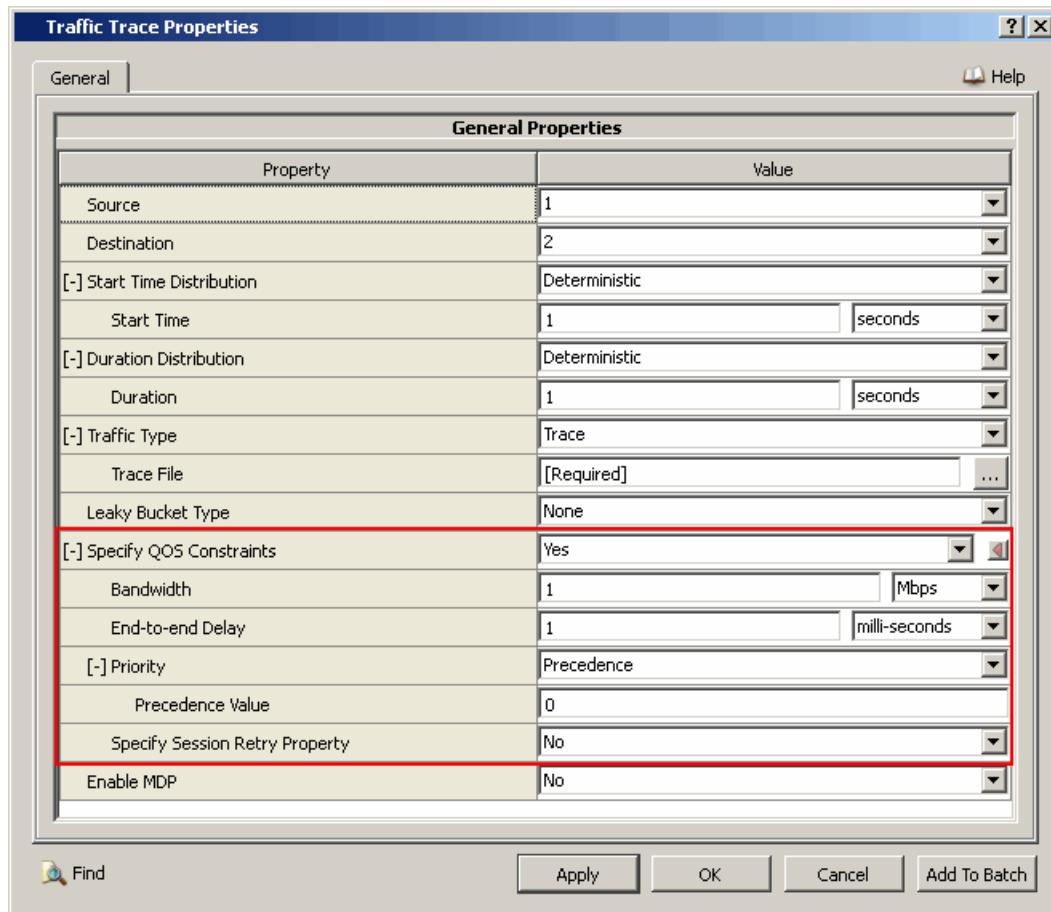


FIGURE 8-63. Setting QoS Parameters

TABLE 8-102. Command Line Equivalent of QoS Parameters

| GUI Parameter | Command Line Parameter |
|--|------------------------|
| Specify QOS Constraints and dependent parameters | <QoS Specification> |

2. Set the dependent parameters of **Specify QOS Constraints** as described in [Section 8.9.3.1.6](#).

8.10.3.1.6 Configuring MDP Parameters

To configure the MDP parameters, perform the following steps:

1. To specify MDP parameters, set **Enable MDP** to Yes; otherwise, set **Enable MDP** to No.

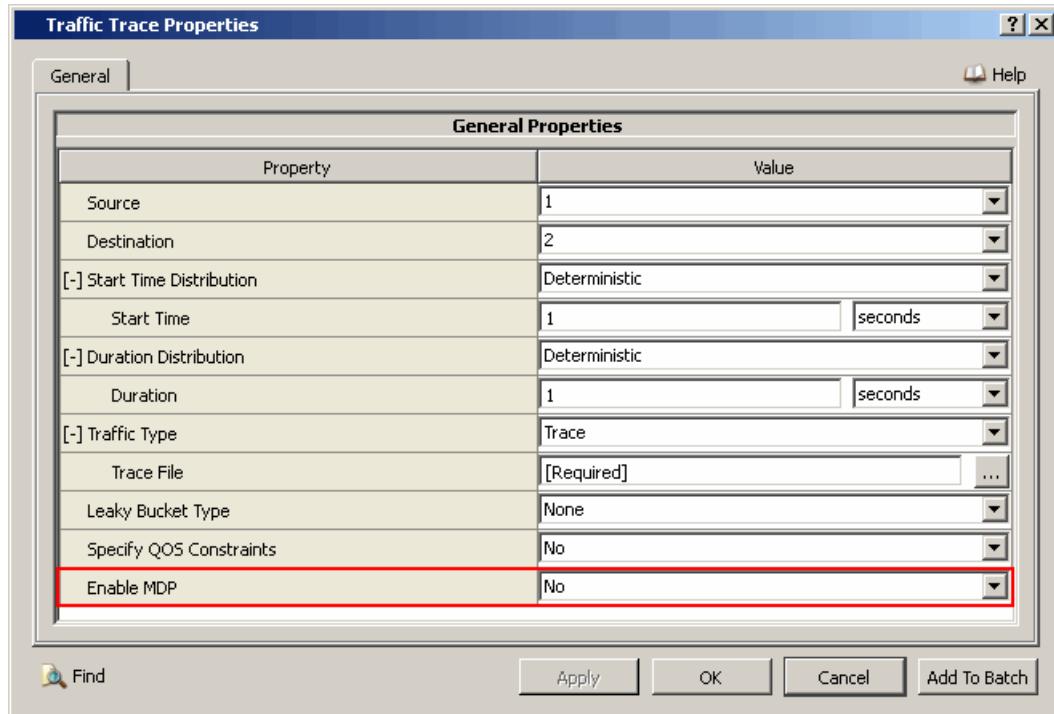


FIGURE 8-64. Enabling MDP

TABLE 8-103. Command Line Equivalent of MDP Parameters

| GUI Parameter | Command Line Parameter |
|-------------------------|------------------------|
| Enable MDP (set to Yes) | MDP - ENABLED |

2. If **Enable MDP** is set to Yes, then set the dependent parameters listed in Table 8-104.

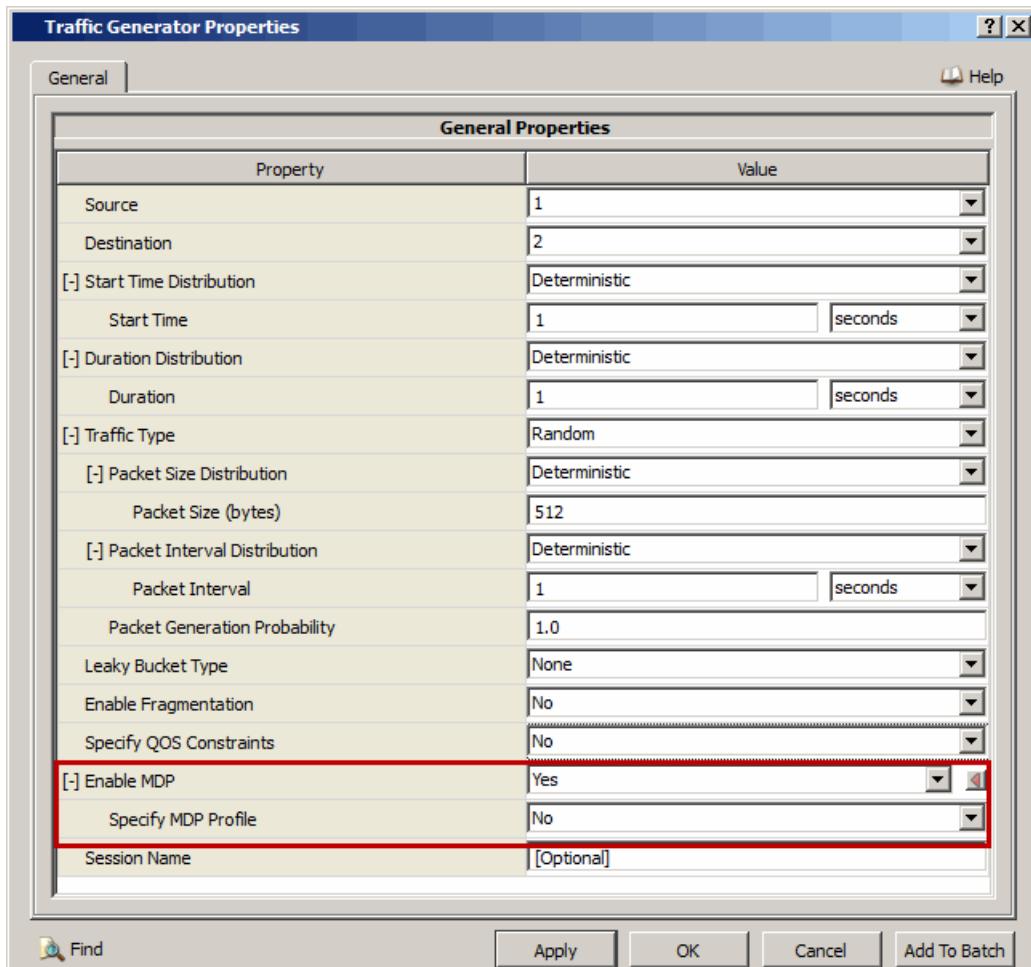


FIGURE 8-65. Configuring MDP Parameters

TABLE 8-104. Command Line Equivalent of MDP Configuration Parameters

| GUI Parameter | Command Line Parameter |
|----------------------------------|------------------------|
| Specify MDP Profile (set to Yes) | MDP - PROFILE |

Setting Parameters

- To specify an MDP profile, set **Specify MDP Profile** to Yes. To use the default MDP profile, set **Specify MDP Profile** to No.

3. If **Specify MDP Profile** is set to Yes, then set the parameters listed in [Table 8-105](#).

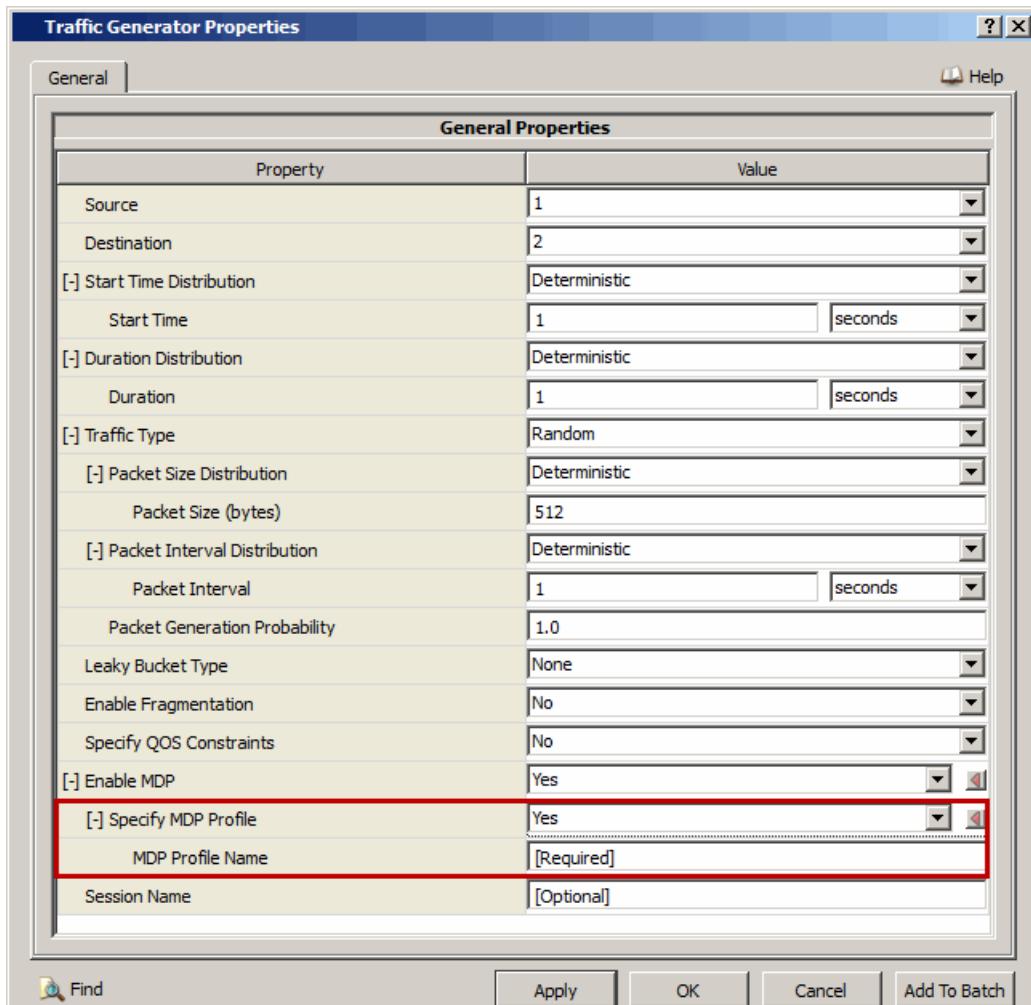


FIGURE 8-66. Specifying MDP Profile

TABLE 8-105. Command Line Equivalent of MDP Profile Parameters

| GUI Parameter | Command Line Parameter |
|------------------|------------------------|
| MDP Profile Name | <profile-name> |

8.10.3.2 Configuring a Single Host Traffic-Trace Session

To configure a single host Traffic-Trace session, perform the following steps:

1. Click the **TRAF TRC** button in the **Single Host Applications** tab of the Standard Toolset.
2. On the canvas, click on the client node.
3. Open the Traffic Generator Properties Editor by doing one of the following:
 - On the canvas, either double-click on the  symbol next to the client node or right-click on the  symbol and select **Properties** from the menu.
 - In the **Applications** tab of Table View, either double-click on the application row or right-click on the application row and select **Properties** from the menu.
4. Set the application parameters as described in [Section 8.10.3.2.1](#) to [Section 8.10.3.2.5](#).
5. Set the statistics parameters as described in [Section 8.10.3.3](#).

8.10.3.2.1 Configuring Source and Destination Parameters

Set the source and destination parameters listed in [Table 8-106](#).

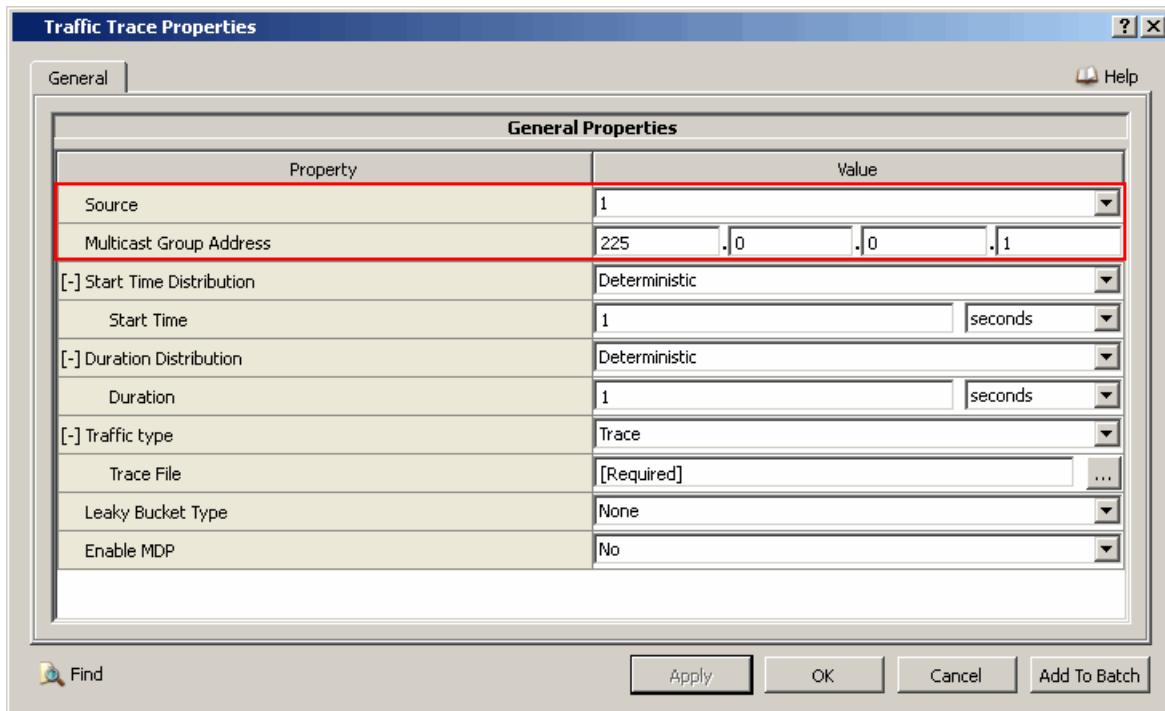


FIGURE 8-67. Setting Source and Destination Parameters

TABLE 8-106. Command Line Equivalent of Source and Destination Parameters

| GUI Parameter | Command Line Parameter |
|-------------------------|------------------------|
| Source | <Source> |
| Multicast Group Address | <Destination> |

Setting Parameters

- To specify an IP address as the source ID, set **Source** to one of the IP addresses listed in the drop-down list.
- Set **Multicast Group Address** to the address of the multicast group that is to receive traffic from the source.

8.10.3.2.2 Configuring Session Parameters

Configure the session parameters as described in [Section 8.10.3.2.2](#).

8.10.3.2.3 Configuring Traffic Parameters

Configure the traffic parameters as described in [Section 8.10.3.2.3](#).

8.10.3.2.4 Configuring Leaky Bucket Parameters

Configure the leaky bucket parameters as described in [Section 8.10.3.2.4](#).

8.10.3.2.5 Configuring MDP Parameters

Configure the MDP parameters as described in [Section 8.10.3.2.5](#).

8.10.3.3 Configuring Statistics Parameters

Statistics for applications (including Traffic-Trace) can be collected at the global and node levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for Traffic-Trace, check the box labeled **Application** in the appropriate properties editor.

TABLE 8-107. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| Application | Global, Node | APPLICATION-STATISTICS |

8.10.4 Statistics

[Table 8-108](#) shows the Traffic-Trace statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 8-108. Traffic-Trace Statistics

| Statistics | Description |
|-----------------------------|---|
| Traffic-Trace Client | |
| Server Address | Address of the server |
| Session Start Time (s) | Time when session started, in seconds |
| Session End Time (s) | Time at which session ended, in seconds |
| Session Status | Current status of the session |
| Total Bytes Sent | Total number of bytes sent |
| Total Data Units Sent | Total number of data units sent |
| Total Data Units Dropped | Total number of data units dropped |
| Throughput (bits/s) | Client side throughput, in bits/second |

TABLE 8-108. Traffic-Trace Statistics (Continued)

| Statistics | Description |
|------------------------------|--|
| Traffic-Trace Server | |
| Client Address | Address of the client |
| Session Start Time (s) | Time in second when session started |
| Session End Time | Time in second when session ended |
| Session Status | Current status of the session |
| Total Bytes Received | Total number of bytes received |
| Total Data Units Received | Total number of data units received |
| Throughput (bits/s) | Server-side throughput, in bits/second |
| Average End-to-End delay (s) | Average end-to-end delay faced for packet transmission, in seconds |
| Average Jitter (s) | Average jitter faced for packet transmission, in seconds |

8.11 Variable Bit Rate (VBR) Traffic Generator

8.11.1 Description

VBR is generally used to fill in background traffic in order to affect the performance of other applications being analyzed, or to simulate the performance of generic multimedia traffic.

The VBR is used for connections that transport traffic at variable rates:

1. Traffic that relies on accurate timing between the traffic source (an example is compressed video streams) and destination
2. Traffic for which there is no inherent reliance on time synchronisation between the traffic source and destination, but there is a need for an attempt at a guaranteed bandwidth or latency (an example is Frame Relay interworking).

8.11.2 Command Line Configuration

Application Configuration File Parameters

To specify VBR traffic, include the following statement in the application configuration (.app) file:

```
VBR <src> <dest> <item-size> <mean-interval>
    <start-time> <end-time>
    [TOS <tos-value> | DSCP <dscp-value> |
     PRECEDENCE <precedence-value>]
    [MDP-ENABLED [MDP-PROFILE <profile-name>]]
    [APPLICATION-NAME <application-name>]
```

Note: All parameters should be entered on the same line.

The VBR parameters are described in [Table 8-109](#). See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 8-109. VBR Parameters

| Parameter | Value | Description |
|--------------------------------|---|---------------------------------|
| <src> <i>Required</i> | Integer or IP Address | Client node's ID or IP address. |
| <dest> <i>Required</i> | Integer or IP Address | Server node's ID or IP address. |
| <item-size> <i>Required</i> | Integer <i>Range:</i> [120, 65023] <i>Unit:</i> bytes | Size of each item. |

TABLE 8-109. VBR Parameters (Continued)

| Parameter | Value | Description |
|--|-----------------------------------|---|
| <mean-interval> <i>Required</i> | Time <i>Range:</i> > 0 | Average time between transmission of items. Inter-packet times are exponentially distributed. |
| <start-time> <i>Required</i> | Time <i>Range:</i> ≥ 0S | Time when the transmission of packets should begin. |
| <end-time> <i>Required</i> | Time <i>Range:</i> ≥ 0S | Time when the transmission of packets should end. The end time must be either greater than start time or equal to 0 |
| TOS <tos-value> <i>Optional</i> | Integer <i>Range:</i> [0, 255] | Value of the 8-bit TOS field of the IP header for the packets generated. See note. |
| DSCP <dscp-value> <i>Optional</i> | Integer <i>Range:</i> [0, 63] | Value of the 6-bit DSCP field of the IP header for the packets generated. See note. |
| PRECEDENCE <precedence-value> <i>Optional</i> | Integer <i>Range:</i> [0, 7] | Value the 3-bit Precedence field of the IP header for the packets generated. See note. |
| MDP-ENABLED <i>Optional</i> | N/A | Keyword which specifies that MDP is enabled for the application. Note: If this keyword is not included, then the application does not run with MDP. |

TABLE 8-109. VBR Parameters (Continued)

| Parameter | Value | Description |
|--|--------|---|
| MDP-PROFILE <i><profile-name></i> <i>Optional</i> | String | <p>Name of the MDP profile to be used with the application.</p> <p>This should be the name of a MDP profile defined in the MDP profile file (see Section 7.2.4.3).</p> <p>This MDP profile takes precedence over the MDP profile associated with the node in the scenario configuration file, if any.</p> <p>Note: This parameter can be included only if the parameter MDP-ENABLED is also included.</p> <p>Note: If parameter MDP-ENABLED is included but parameter MDP-PROFILE is not included, and a MDP profile is specified for the node in the scenario configuration file, then the MDP profile specified for the node is used.</p> <p>If parameter MDP-ENABLED is included but parameter MDP-PROFILE is not included, and a MDP profile is not specified for the node in the scenario configuration file, then default MDP values for the MDP profile parameters are used (see Section 7.2.4.3).</p> |
| APPLICATION-NAME <i><application-name></i> <i>Optional</i> | String | Name of the VBR session. This name is printed in the statistics file and statistics database. |

Note: At most one of the three parameters PRECEDENCE, DSCP, and TOS can be specified. If PRECEDENCE, DSCP or TOS is not specified, PRECEDENCE 0 is used as default.

Examples of Parameter Usage

1. In the following example of VBR configuration, node 1 sends node 2 items of 2048 bytes each, starting at 50 seconds into the simulation and up to at 500 minutes into the simulation. The mean inter-packet time is 1 minute.

```
VBR 1 2 2048 1M 50S 500M
```

2. This is the same as the previous example, except that VBR runs with MDP and uses the MDP profile associated with node 1. If there is no MDP profile associated with node 1, default values for MDP profile parameters are used.

```
VBR 1 2 2048 1M 50S 500M MDP-ENABLED
```

3. This is the same as the previous example, except that MDP uses the user-defined MDP profile profile-1.

```
VBR 1 2 2048 1M 50S 500M MDP-ENABLED MDP-PROFILE profile-1
```

8.11.3 GUI Configuration

VBR is configured using the VBR Properties Editor.

To configure a VBR session between two nodes, perform the following steps:

1. Click the **VBR** button in the **Applications** tab of the Standard Toolset.
2. On the canvas, click on the source node, drag the mouse to the destination node, and release.
3. Open the VBR Properties Editor by doing one of the following:
 - c. Right-click in the application link on the canvas and select **Properties** from the menu.
 - d. In the **Applications** tab of Table View, either double-click on the application row or right-click on the application row and select **Properties** from the menu.
4. Set the parameters listed in Table 8-110.

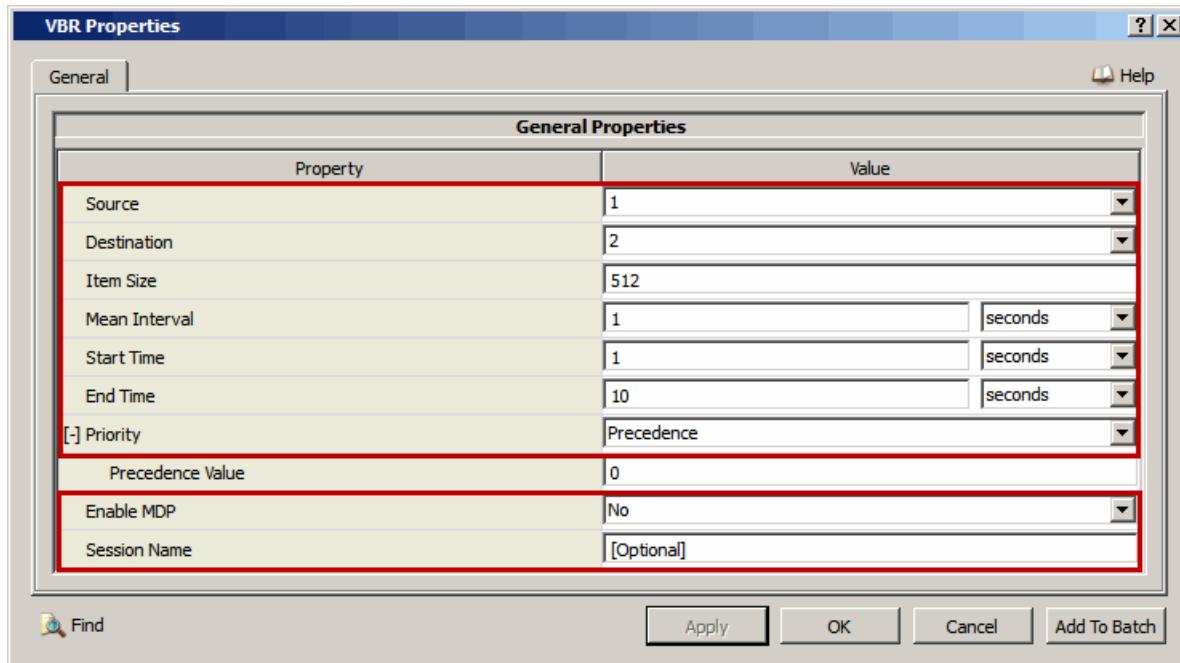


FIGURE 8-68. Setting VBR Parameters

TABLE 8-110. Command Line Equivalent of VBR Parameters

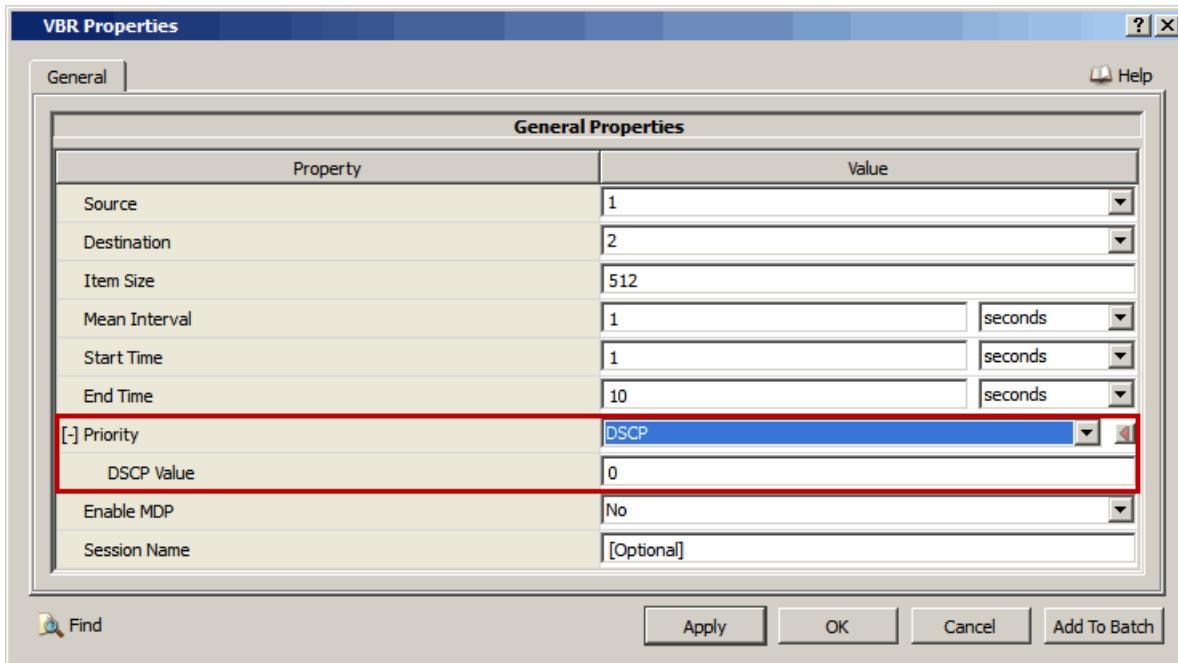
| GUI Parameter | Command Line Parameter |
|------------------------------|------------------------|
| Source | <src> |
| Destination | <dest> |
| Item Size | <item-size> |
| Mean Interval | <mean-interval> |
| Start Time | <start-time> |
| End Time | <end-time> |
| Priority (set to DSCP) | DSCP |
| Priority (set to Precedence) | PRECEDENCE |

TABLE 8-110. Command Line Equivalent of VBR (Continued)Parameters (Continued)

| GUI Parameter | Command Line Parameter |
|-------------------------|-------------------------------------|
| Priority (set to TOS) | TOS |
| Enable MDP (set to Yes) | MDP-ENABLED |
| Session Name | APPLICATION-NAME <application-name> |

Setting Parameters

- To specify an IP address as the source (destination) ID, set **Source (Destination)** to one of the IP addresses listed in the drop-down list.
 - To set DSCP, Precedence, or TOS bits for packets, set **Priority** to *DSCP*, *Precedence*, or *TOS*, respectively.
 - To enable MDP, set **Enable MDP** to Yes.
5. If **Priority** is set to *DSCP*, *Precedence*, or *TOS*, set the appropriate dependent parameter listed in Table 8-111. Figure 8-69 shows how to set the dependent parameter when **Priority** is set to *DSCP*. Setting dependent parameters for the other two options is similar.

**FIGURE 8-69. Setting DSCP Bits****TABLE 8-111. Command Line Equivalent of Priority Parameters**

| GUI Parameter | Command Line Parameter |
|------------------|------------------------|
| DSCP Value | <dscp-value> |
| Precedence Value | <precedence-value> |
| TOS Value | <tos-value> |

6. If **Enable MDP** is set to Yes, then set the parameters listed in [Table 8-112](#).

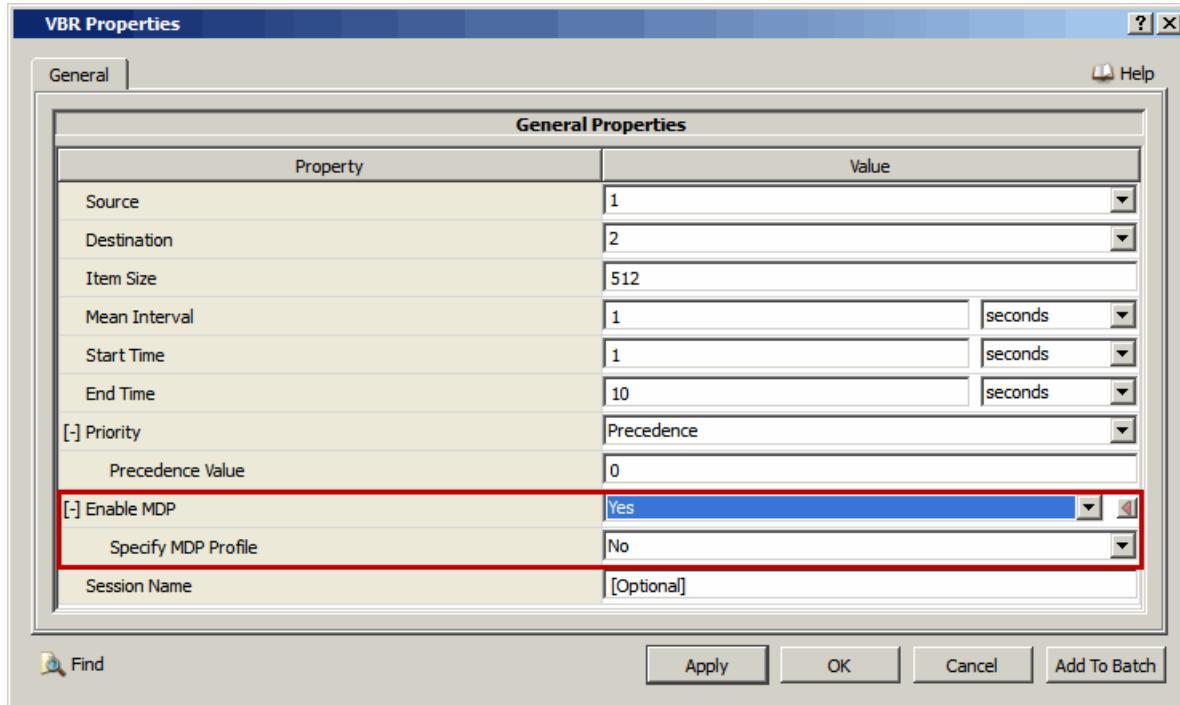


FIGURE 8-70. Enabling MDP

TABLE 8-112. Command Line Equivalent of MDP Parameters

| GUI Parameter | Command Line Parameter |
|----------------------------------|------------------------|
| Specify MDP Profile (set to Yes) | MDP-PROFILE |

Setting Parameters

- To specify an MDP profile, set **Specify MDP Profile** to Yes. To use the default MDP profile, set **Specify MDP Profile** to No.

7. If **Specify MDP Profile** is set to Yes, then set the parameters listed in [Table 8-113](#).

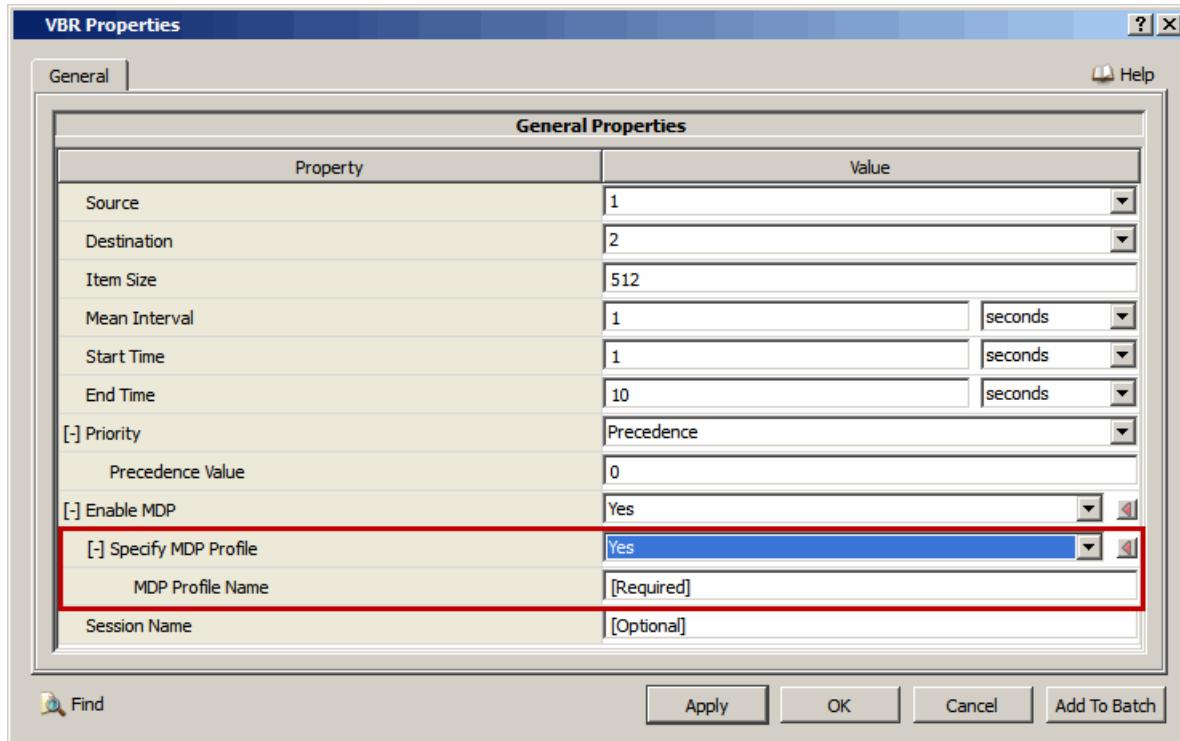


FIGURE 8-71. Specifying MDP Profile

TABLE 8-113. Command Line Equivalent of MDP Profile Parameters

| GUI Parameter | Command Line Parameter |
|------------------|------------------------|
| MDP Profile Name | <profile-name> |

Configuring Statistics Parameters

Statistics for applications (including VBR) can be collected at the global and node levels. See Section 4.2.9 of *QualNet User's Guide* for details of configuring statistics parameters.

To enable statistics collection for VBR, check the box labeled **Application** in the appropriate properties editor.

TABLE 8-114. Command Line Equivalent of Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|---------------|------------------------|------------------------|
| Application | Global, Node | APPLICATION-STATISTICS |

8.11.4 Statistics

This section describes the file, database, and dynamic statistics of the VBR model.

8.11.4.1 File Statistics

[Table 8-115](#) lists the VBR statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 8-115. VBR Statistics

| Statistic | Description |
|--|--|
| Total Bytes Sent | Total number of bytes sent. |
| Total Bytes Received | Total number of bytes received. |
| VBR Client | |
| Unicast Session Start (seconds) | Time in seconds, when first unicast session was started. |
| Unicast Session Finish (seconds) | Time in seconds, when first unicast session was finished. |
| First Unicast Fragment Sent (seconds) | Time in seconds, when first unicast fragment was sent. |
| Last Unicast Fragment Sent (seconds) | Time in seconds, when last unicast fragment was sent. |
| First Unicast Fragment Received (seconds) | Time in seconds, when first unicast fragment was received. |
| Last Unicast Fragment Received (seconds) | Time in seconds, when last unicast fragment was received. |
| Total Unicast Fragments Sent (fragments) | Total number of unicast fragments sent. |
| Total Unicast Fragments Received (fragments) | Total number of unicast fragments received. |
| First Unicast Message Sent (seconds) | Time in seconds, when first unicast message was sent. |
| Last Unicast Message Sent (seconds) | Time in seconds, when last unicast message was sent. |
| First Unicast Message Received (seconds) | Time in seconds, when first unicast message was received. |
| Last Unicast Message Received (seconds) | Time in seconds, when last unicast message was received. |
| Total Unicast Messages Sent (messages) | Total number of unicast messages sent. |
| Total Unicast Messages Received (messages) | Total number of unicast messages received. |
| Total Unicast Data Sent (bytes) | Total number of unicast data bytes sent. |
| Total Unicast Data Received (bytes) | Total number of unicast data bytes received. |
| Total Unicast Overhead Sent (bytes) | Total number of unicast overhead bytes sent. |
| Total Unicast Overhead Received (bytes) | Total number of unicast overhead bytes received. |
| Average Unicast End-to-End Delay (seconds) | Average unicast end-to-end delay. |
| Unicast Offered Load (bits/second) | Unicast offered load. |

TABLE 8-115. VBR Statistics (Continued)

| Statistic | Description |
|--|--|
| Unicast Received Throughput (bits/second) | Unicast received throughput. |
| Smoothed Unicast Jitter (seconds) | Smoothed unicast jitter. |
| Average Unicast Jitter (seconds) | Average unicast jitter. |
| Total Unicast Jitter (seconds) | Total unicast jitter. |
| VBR Server | |
| Unicast Session Start (seconds) | Time in seconds, when first unicast session was started. |
| Unicast Session Finish (seconds) | Time in seconds, when first unicast session was finished. |
| First Unicast Fragment Sent (seconds) | Time in seconds, when first unicast fragment was sent. |
| Last Unicast Fragment Sent (seconds) | Time in seconds, when last unicast fragment was sent. |
| First Unicast Fragment Received (seconds) | Time in seconds, when first unicast fragment was received. |
| Last Unicast Fragment Received (seconds) | Time in seconds, when last unicast fragment was received. |
| Total Unicast Fragments Sent (fragments) | Total number of unicast fragments sent. |
| Total Unicast Fragments Received (fragments) | Total number of unicast fragments received. |
| First Unicast Message Sent (seconds) | Time in seconds, when first unicast message was sent. |
| Last Unicast Message Sent (seconds) | Time in seconds, when last unicast message was sent. |
| First Unicast Message Received (seconds) | Time in seconds, when first unicast message was received. |
| Last Unicast Message Received (seconds) | Time in seconds, when last unicast message was received. |
| Total Unicast Messages Sent (messages) | Total number of unicast messages sent. |
| Total Unicast Messages Received (messages) | Total number of unicast messages received. |
| Total Unicast Data Sent (bytes) | Total number of unicast data bytes sent. |
| Total Unicast Data Received (bytes) | Total number of unicast data bytes received. |
| Total Unicast Overhead Sent (bytes) | Total number of unicast overhead bytes sent. |
| Total Unicast Overhead Received (bytes) | Total number of unicast overhead bytes received. |
| Average Unicast End-to-End Delay (seconds) | Average unicast end-to-end delay. |
| Unicast Offered Load (bits/second) | Unicast offered load. |
| Unicast Received Throughput (bits/second) | Unicast received throughput. |
| Smoothed Unicast Jitter (seconds) | Smoothed unicast jitter. |
| Average Unicast Jitter (seconds) | Average unicast jitter. |
| Total Unicast Jitter (seconds) | Total unicast jitter. |

8.11.4.2 Database Statistics

In addition to the file statistics, the VBR model also enters statistics in various scenario statistics database tables. Refer to *QualNet Statistics Database User's Guide* for details.

8.11.4.3 Dynamic Statistics

The following dynamic statistics are enabled for the VBR model (refer to Chapter 5 of *QualNet User's Guide* for details of viewing dynamic statistics in the GUI during the simulation):

- Total Bytes Sent
- Total Bytes Received

9

Multi-layer Models

This chapter describes features, configuration requirements and parameters, statistics, and scenarios for Multi-layer Models, and consists of the following sections:

- Asynchronous Transfer Mode (ATM)

9.1 Asynchronous Transfer Mode (ATM)

ATM is based on the following documents:

- RFC 2225 “Classical IP and ARP over ATM” M. Laubach, J. Halpern. April 1998.
- RFC 2684 “Multiprotocol Encapsulation over ATM Adaptation Layer 5” D. Grossman, J. Heinanen. September 1999
- ATM Forum Addressing Specification: Reference Guide AF-RA-0106.000.

9.1.1 Description

Asynchronous Transfer Mode (ATM) is an International Telecommunication Union-Telecommunications (ITU-T) standard for a connection-oriented cell relay protocol. Information bit streams are conveyed in small fixed-size cells (53 bytes). The ATM library in QualNet has implemented the ATM layer 2, ATM signaling, and ATM Adaptation Layer Type 5 (AAL5). It also supports interoperability between IP networks and ATM networks known as IP over ATM. You can use the ATM library to simulate pure ATM backbone networks as well as IP over ATM networks.

9.1.1.1 ATM Backbone

ATM backbone network consists of ATM switches and ATM end system nodes. They are connected via wired point-to-point links.

9.1.1.1.1 ATM Adaptation Layer (AAL)

The ATM Adaptation Layer supports mapping between the ATM layer 2 and the next higher layer, and performs user-required functions such as control and management planes. It supports non-assured transfer of user data frames. The AAL type 5 is characterized by the fact that in most of the cells there is no overhead encountered. The AAL type 5 provides the capabilities to transfer the AAL-SDU from one AAL SAP to one another AAL SAP through the ATM network in the following manner:

Starting from the ATM cell stream on a single VCC (Virtual Channel connection), the only overhead the SAR sub layer uses is the payload type field in the last cell of a sequence of cells corresponding to a single PDU. A non-zero value of the AAL_Indicate field identifies the last cell in the sequence of cells indicating that the receiver can begin reassembly. The SAR sub layer reassembles the CPCS_PDU and passes it to the CPCS sub layer, which first uses the CRC-32 field to check for any errors in the received PDU. Normally, CPCS discards corrupted PDUs, but may optionally deliver them onto an SSCS. The CPCS removes the PAD and other trailer fields before passing the AAL-SDU across the AAL5-SAP. The transmit operation is the reverse of the above. Depending on the payload size (may be any integer number of octets in the range of 1 to $2^{16}-1$), PAD field is chosen as a variable length so that the entire CPCS-PDU is an exact multiple of 48 and can be directly segmented into cell payloads. The length field identifies the length of the CPCS-PDU payload so that the receiver can remove the PAD field. Since 16 bits are allocated to the length field, the maximum payload length is $2^{16}-1= 65535$ octets.

9.1.1.1.2 ATM Layer 2

ATM layer 2 is in between Physical layer and AAL layer. This is the most important layer of the ATM protocol stack. Most of the essential functionalities are performed at this layer. It receives 48 byte packets from AAL, attaches a five-byte header and sends it to the lower layer. Apart from the multiplexing and demultiplexing of cell stream, the translation of the connection identifiers (VPI / VCI) and cell switching is done at this layer at the intermediate nodes. This layer also has the mechanisms for traffic management and connection admission control. It checks for resource availability on receiving a request for connection establishment and checking the data rate offered by the user on a connection to the network.

9.1.1.1.3 ATM Signaling

Signaling procedure specifies the valid sequence of message exchanged between the user & the network, the rules for verifying consistency of the parameters, and the actions taken to establish and release ATM layer connections. There are various issues that govern the design of an efficient & effective signaling protocol such as acknowledgements, timer protection, parameter negotiation, etc.

Transmission media, being unreliable by nature, need a two-way or three-way hand shaking for a reliable communication. *Timer* is used to avoid inordinate delays in case the signaling message gets lost or corrupted. *Parameter negotiation* refers to the process of fixing connection parameter through the exchange of signaling information. Two end systems can have more than one active connection simultaneously. To identify each call separately in such a scenario, you must introduce the *Call Identification process*. Generally, a call goes through three distinct phases:

1. **Call establishment phase:** Through the two-way or three-way handshaking process, each end system is informed about the current call and status of the other to make them ready for the data transfer process.
2. **Data transfer phase:** The actual data transfer takes place in this phase.
3. **Call clearing (releasing) phase:** As the call is cleared in this phase, all resources allocated for the call are freed-up.

9.1.1.2 IP over ATM

In QualNet, the interoperability between ATM backbone and IP subnets is implemented following the classical IP over ATM specifications. You can forward IP packets through ATM backbones. It introduces the concept of a logical entity called Logical IP Subnet (LIS) which treats the ATM network as a number of separate IP subnets connected through routers. Membership to an LIS is defined by software configuration. Transportation of various protocol data units over an ATM network uses the following functionalities:

- Addressing
- Address resolution
- Data encapsulation
- Routing

9.1.1.2.1 Addressing

In ATM Forum Specifications the term address is used to mean locator (for example, address indicates the location of an interface). An individual ATM address is 20 octets long and identifies the location of a single ATM interface. Basically the total address consists of following parts in this format.

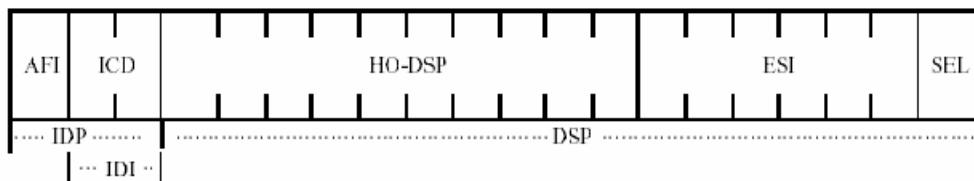


FIGURE 9-1. ATM Address Format

The Initial Domain Part (IDP) uniquely specifies an administration authority, which has the responsibility for allocating and assigning values of the Domain Specific Part (DSP). QualNet implements only the International Code Designator (ICD) Address format. International organizations use this format for address allocation.

9.1.1.2.2 Address Resolution

For operation of IP over ATM, a mechanism is used to resolve IP addresses to the corresponding ATM addresses within an ATM logical IP subnet. It uses ATMARP and InATMARP protocols for this purpose. All nodes within the LIS are configured with the unique ATM address of the ATMARP server, which is used to resolve the requests of the LIS clients to map the IP address to the ATM address. At present no ATMARP Server is configured separately, rather, a static table is provided with all ARP related information and an address table is kept at the end systems for this purpose.

9.1.1.2.3 Data Encapsulation

IETF describes two encapsulation methods for carrying network interconnect traffic over the ATM AAL5. The first of these two methods, LLC/SNAP Encapsulation, allows the transport network and link layer packets across an ATM connection. The second method of encapsulation, VC Multiplexing, allows the multiplexing of multiple packet types on the same connection. These methods allow you to conserve connection resource space (all data transfer between two nodes across the same connection) and to save connection latency after the first connection setup.

- **LLC/SNAP Encapsulation** - This encapsulation method is used for carrying multiple protocols over a single ATM VC connection. An LLC header placed in front of the carried Protocol Data Unit (PDU) identifies the packet type (IP, IPX, and AppleTalk). This approach is preferred when separate VCs for each carried protocol are either expensive or not possible (for example, the only connection type supported is PVC or charging is based on the number of VC allocated). This method is the default encapsulation method for all IP over ATM protocols.
- **VC Multiplexing** - In this method, each protocol is carried over a separate ATM VC, with the type of protocol identified at connection setup. With VC Based Multiplexing, the data is encapsulated into the CPCS-PDU field of AAL5 directly, so each connection can only carry one protocol. This approach is preferred when you wish to establish a large number of VCs in a fast and economical way.

9.1.1.2.4 Routing

Routing is another important aspect of carrying IP Protocol Data Unit over ATM. In a mixed topology of IP (connectionless), and ATM (connection oriented), paths among various routers can be created dynamically, but there is some difficulty related to QoS. An alternative is to create such a path through management action so that traffic engineering can be accomplished. This leads to the concept of Gateway. Different IP clouds are connected to ATM backbone through their respective gateways, which act as the entry point to other (next Hop) networks (for example, ATM cloud). To reach another IP network, the egress end systems are discovered in SVC. Setup messages are forwarded to all the end systems attached in that logical network. The end system with the proper outgoing route replies back with an alert message to the desired IP. Once the path is established, data can be transferred only through that path.

9.1.2 Features and Assumptions

This section describes the implemented features, omitted features, assumptions and limitations of the ATM model.

9.1.2.1 Implemented Features

- Cell construction
- Cell reception and header validation
- Cell relaying, forwarding and copying
- Cell multiplexing and demultiplexing.
- Interpretation-only data cell and signaling cell
- Explicit forward congestion indication
- Point to point connection

- Connection assignment and removal
- Bandwidth is reserved for each application. Total link bandwidth is shared among all the applications. If bandwidth requirements exceed available bandwidth, no virtual connection is established for that application. One virtual path is reserved for signaling.
- The virtual path setup process for each application is done dynamically, for example, using the SVC process.
- For SVC, connections are refreshed periodically and if a virtual circuit remains idle for a certain period (timeout-time), it is considered as timeout and bandwidth is freed and can be reused.
- IP over ATM features
 - IP Protocol Data Units are carried over the ATM cloud through the gateways.
 - Gateways can be configured from the configuration file for each node present in IP clouds.
 - ATM-ARP is partially implemented for address discovery. Addresses are translated statically.
- The following applications (running at IP nodes) can go through the ATM backbone: CBR, VBR, VOIP, Super-Application, FTP, FTP/Generic, HTTP, Lookup, TELNET, Traffic-Gen, and Traffic-Trace (without QoS specification).

9.1.2.2 Omitted Features

- QoS features. All applications use DEFAULT-SAAL-BW.
- Point to multiple connection
- IPv6 over ATM
- IP Multicast over ATM
- PNNI Routing
- PVC

9.1.2.3 Assumptions and Limitations

- Routing within the ATM clouds is done statically. Static route file is provided externally during configuration.
- Only the CBR application is supported for standalone ATM networks.

9.1.3 Command Line Configuration

This section describes the command line configuration for standalone ATM backbone and for IP over ATM networks.

9.1.3.1 Configuring the ATM Backbone

An ATM network (consisting of ATM switches, ATM end systems, and ATM links) is configured using special keywords which are described below. These parameters should be included in the scenario configuration (.config) file.

- All nodes in the ATM network must be explicitly defined as ATM nodes by using the following parameter declaration:

```
<node list> ATM-NODE      YES
```

where

```
<node list>      Space-separated list of node IDs, enclosed in square brackets, [ and ].
```

- Each ATM node must be designated as an ATM end-node or an ATM switch. If an ATM node is to interface with non-ATM networks or if it is to run an application, then it should be designated as an ATM end-node. An ATM switch can interface only with ATM nodes and can not run applications. ATM end-nodes are defined by using the following parameter declaration:

```
<node-list> ATM-END-SYSTEM      YES
```

where

`<node-list>` Space-separated list of node IDs, enclosed in square brackets, [and].

Note: An ATM node that is not explicitly designated as an ATM end-node is treated as an ATM switch.

- ATM supports only wired point-to-point links. These links are dedicated, error-free, and support the maximum bandwidth in both directions simultaneously. To create an ATM link between two ATM nodes, use the following parameter:

```
ATM-LINK <link-address> {<node-1>, <node-2>}
```

where

`<link-address>` Subnet address of the link in QualNet ATM address format.

`<node-1>, <node-2>` Node IDs of the nodes at the end points of the link.

- Each ATM end node must be made part of a logical subnet. There can be more than one logical subnets. Logical subnets are defined by using the following parameter:

```
[<node-list>] ATM-LOGICAL-SUBNET <subnet-address> <member-nodes>
```

where

`<node-list>` Space-separated list of node IDs, enclosed in square brackets, [and].

The node IDs in `<node-list>` are the same as the nodes IDs in `<member-nodes>`.

`<node-list>` is optional and can be omitted if there is only one logical subnet.

If there are two or more logical subnets, then at most one of them can be defined without `<node-list>` and the others must be defined with `<node-list>`.

`<subnet-address>` IPv4 address of the subnet in QualNet N syntax.

`<member-nodes>` Comma-separated list of node IDs, enclosed in braces, { and }. A range of nodes can also be specified by using the keyword `thru`.

ATM Configuration Parameters

Table 9-1 shows the configurable parameters for ATM. See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 9-1. ATM Configuration Parameters

| Parameter | Value | Description |
|--|---|---|
| ATM Layer 2 Parameters | | |
| ATM-LAYER2-LINK-BANDWIDTH <i>Required</i> Scope: All | Integer <i>Range:</i> ≥ 0 <i>Unit:</i> bps | This parameter specifies the bandwidth of the ATM link. |
| ATM-LAYER2-LINK-PROPAGATION-DELAY <i>Required</i> Scope: All | Time <i>Range:</i> $\geq 0\text{ s}$ | This parameter specifies the propagation delay for wired point-to-point ATM link. |
| ATM Adaptation Layer Parameters | | |
| ATM-CONNECTION-REFRESH-TIME <i>Optional</i> Scope: All | Time <i>Range:</i> $\geq 0\text{ s}$ <i>Default:</i> 5M | This parameter specifies the time after which the SVC connection and the connection table are refreshed. |
| ATM-CONNECTION-TIMEOUT-TIME <i>Optional</i> Scope: All | Time <i>Range:</i> $\geq 0\text{ s}$ <i>Default:</i> 1M | This parameter specifies the time till which the virtual path can remain idle. After that, it is taken as a timeout, and bandwidth is freed. |
| ADAPTATION-PROTOCOL <i>Required</i> Scope: Global, Node | List: • AAL5 | AAL5 is implemented as the Adaptation Layer protocol for ATM. |
| ATM Routing Parameters | | |
| ATM-STATIC-ROUTE <i>Required</i> Scope: Global | List: • YES | This parameter specifies whether or not ATM uses static routes. As ATM routes the packet using static route only this parameter should be set to YES. |
| ATM-STATIC-ROUTE-FILE <i>Required</i> Scope: Global | Filename | Name of the ATM static routes file. The format of the ATM static routes is described in Section 9.1.3.1.1 . |

TABLE 9-1. ATM Configuration Parameters (Continued)

| Parameter | Value | Description |
|--|--|---|
| ATM Queue And Scheduler Parameters | | |
| ATM-RED-MIN-THRESHOLD <i>Optional</i> Scope: All | Integer <i>Range:</i> ≥ 0 <i>Default:</i> 5 | Number of packets in the queue that represents the lower bound at which packets can be randomly dropped or marked. |
| ATM-RED-MAX-THRESHOLD <i>Optional</i> Scope: All | Integer <i>Range:</i> ≥ 0 <i>Default:</i> 15 | Number of packets in the queue that represents the upper bound at which packets can be randomly dropped or marked. Note: ATM-RED-MAX-THRESHOLD should always be greater than or equal to ATM-RED-MIN-THRESHOLD. |
| ATM-RED-MAX-PROBABILITY <i>Optional</i> Scope: All | Real <i>Range:</i> [0.0, 1.0] <i>Default:</i> 0.02 | Maximum probability at which a packet can be dropped (before the queue is completely full). |
| ATM-RED-SMALL-PACKET-TRANSMISSION-TIME <i>Optional</i> Scope: All | Time <i>Range:</i> $\geq 0\text{S}$ <i>Default:</i> 10MS | Typical amount of time that it would take to transmit a small packet. This is used to estimate the queue average length during idle periods. |
| ATM-QUEUE-SIZE <i>Optional</i> Scope: All Instances: queue number | Integer <i>Range:</i> ≥ 0 <i>Default:</i> 15000 <i>Unit:</i> bytes | ATM Queue uses a special type of Random Early Detection (RED) queue internally. RED marks packets; it does not drop them when queue is not full. |
| ATM Statistics Parameters | | |
| ATM-LAYER2-STATISTICS <i>Optional</i> Scope: All | List: • YES • NO <i>Default:</i> NO | Enables the collection of ATM Layer2 specific statistics. |
| ATM-SIGNALLING-STATISTICS <i>Optional</i> Scope: Global, Node | List: • YES • NO <i>Default:</i> NO | Enables the collection of ATM signaling statistics. |

TABLE 9-1. ATM Configuration Parameters (Continued)

| Parameter | Value | Description |
|---|---|--|
| ADAPTATION-LAYER-STATISTICS <i>Optional</i> Scope: Global, Node | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Enables the collection of ATM Adaptation Layer statistics. |
| ATM-SCHEDULER-STATISTICS <i>Optional</i> Scope: All | List: <ul style="list-style-type: none">• YES• NO <i>Default:</i> NO | Enables the collection of ATM scheduler statistics. |

9.1.3.1.1 Format of the ATM Static Routes File

Each line in the ATM static routes file has the following format:

```
<node ID> <dest address> <outgoing interface>
```

where

- | | |
|----------------------|--|
| <node ID> | Node identifier. |
| <dest address> | Destination address in ATM cloud. This can be either a host IP address or a network IP address. |
| <outgoing interface> | IP address of the outgoing interface. |

9.1.3.2 Configuring IP over ATM

To configure an IP over ATM network scenario, the following steps are required in addition to the ATM backbone configuration described in [Section 9.1.3.1](#).

1. Connect IP subnets to ATM end systems by point-to-point links or wired or wireless subnets. End system nodes should be placed in a logical IP subnet.
2. The ARP module should be configured to provide the capability to translate logical IP addresses to ATM addresses of ATM interfaces. ARP should be configured to use static ARP cache file.

See [Section 2.4](#) for details of configuring ARP.

- Note:**
1. The <Protocol> entry must be an IP address
 2. For IP over ATM scenarios, the <hardware type> entry must be ATM and the <timeout> entry must be 0 in each line of the ARP static cache file (see [Section 2.4.4.1](#))

Example

The following is an example of the ARP static cache file for an IP over ATM scenario:

```
1      IP      192.168.5.1      ATM      23:4f:5C:aa:FE:00      0
2      IP      192.168.5.2      ATM      23:4f:5C:aa:FE:01      0
```

3. To enable routing of packets, each IP subnet must configure a default gateway that routes packets to the gateway node connecting the IP subnet to the ATM backbone. Some of the ATM end systems are configured as the gateway for associated IP clouds. These nodes are responsible for transferring data from IP to ATM clouds and vice versa. The format for configuring a default gateway is:

```
[<node ID>] DEFAULT-GATEWAY <gateway-address>
```

<node ID> Node ID of the node for which the default gateway is configured.

<gateway-address> Node ID or the IP address of the gateway node.

9.1.4 GUI Configuration

This section describes how to configure ATM networks in the GUI.

9.1.4.1 Configuring ATM Networks

This section describes how to set up an ATM network.

ATM Devices

The ATM device can be selected from the Devices toolbar of the Standard Toolset.

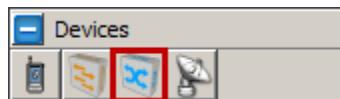


FIGURE 9-2. ATM Device Button in Devices Toolbar

ATM End Systems

To configure an ATM device as an end system, perform the following steps:

1. Go to **ATM Device Properties Editor > General**.
2. Set **Node Type** to *ATM End System*.

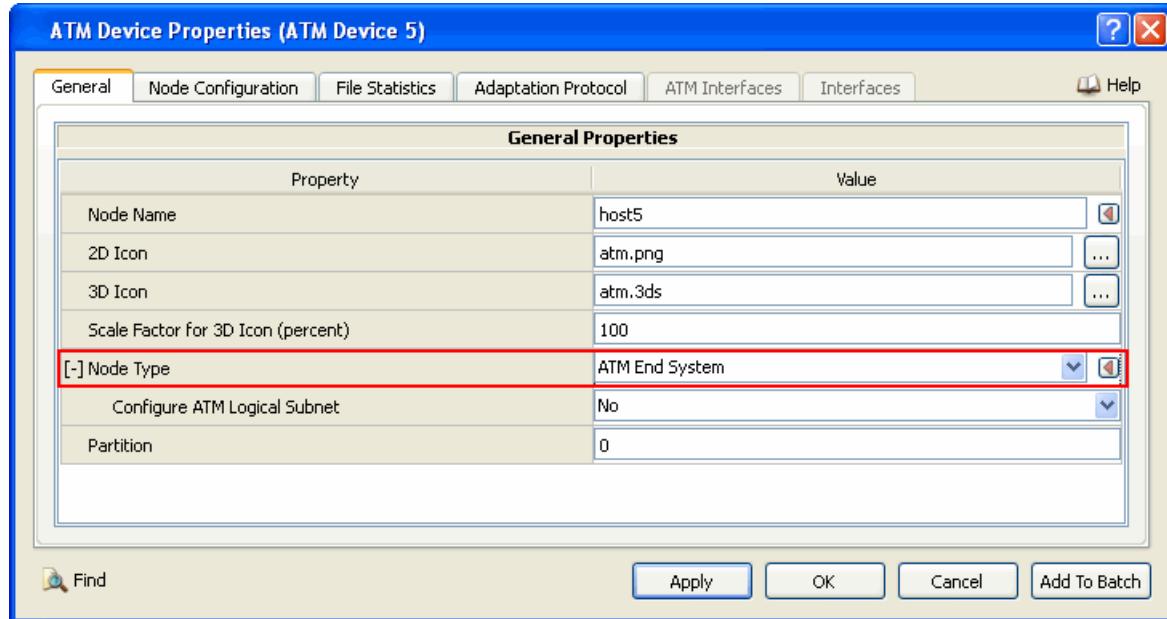


FIGURE 9-3. Setting ATM End System

ATM Links

To create an ATM Link, select the Link button in the Links toolbar, and create a link between two ATM switches.

Configuring ATM Logical Subnet

ATM logical subnets can be defined at the global (scenario) level or at the node level. If there is only one logical subnet, it is recommended that it be configured at the scenario level. If there are multiple logical subnets, it is recommended that they be configured at the node level (although one of them can be configured at the scenario level).

To configure an ATM logical subnet at the scenario level, perform following steps:

1. Go to **Scenario Properties Editor > General > ATM Configuration**.
2. Set **Configure ATM Logical Subnet** to Yes.
3. In the **ATM Logical Subnet** field, specify the subnet address followed by the node IDs of all ATM end systems that belong to this ATM logical subnet.

Note: The node IDs should be separated by commas and the list should be enclosed in { and }.

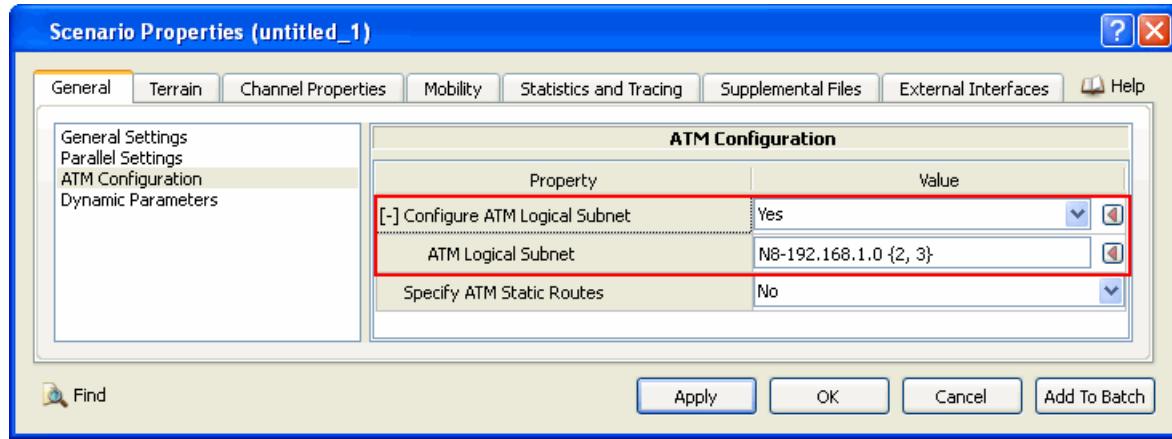


FIGURE 9-4. Configuring ATM Logical Subnet at Scenario Level

To configure an ATM logical subnet at the node level, perform following steps:

1. Go to **ATM Device Properties Editor > General**.
2. Set **Node Type** to *ATM End System*.
3. Set **Configure ATM Logical Subnet** to *Yes*.
4. In the **ATM Logical Subnet** field, specify the subnet address to which this node belongs followed by the node IDs of all ATM end systems that belong to this ATM logical subnet.

Note: The node IDs should be separated by commas and the list should be enclosed in { and }.

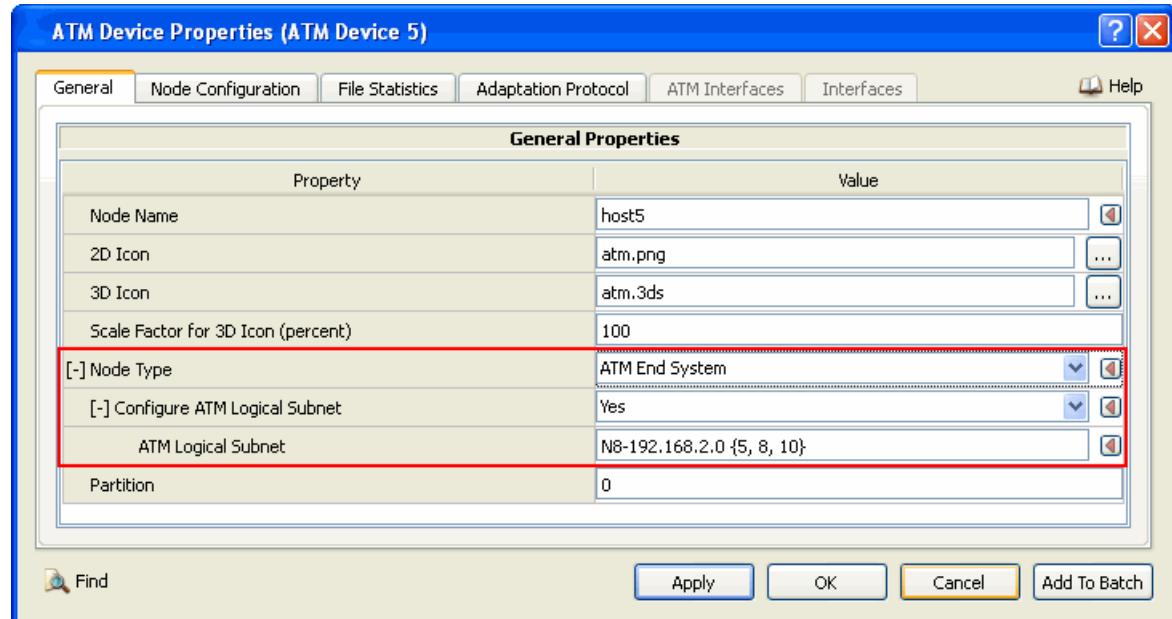


FIGURE 9-5. Configuring ATM Logical Subnet at Node Level

9.1.4.2 Configuring ATM Link Parameters

Section 9.1.4.2.1 describes how to configure properties for symmetric ATM links. Section 9.1.4.2.2 describes how to configure properties for asymmetric ATM links.

9.1.4.2.1 Symmetric ATM Links

To configure link parameters for a symmetric ATM link, perform the following steps:

1. Go to **ATM Link Properties Editor > General**.
2. To configure the link as a symmetric link, set **ATM Symmetric Link** to **Yes**; otherwise, set **ATM Symmetric Link** to **No**.

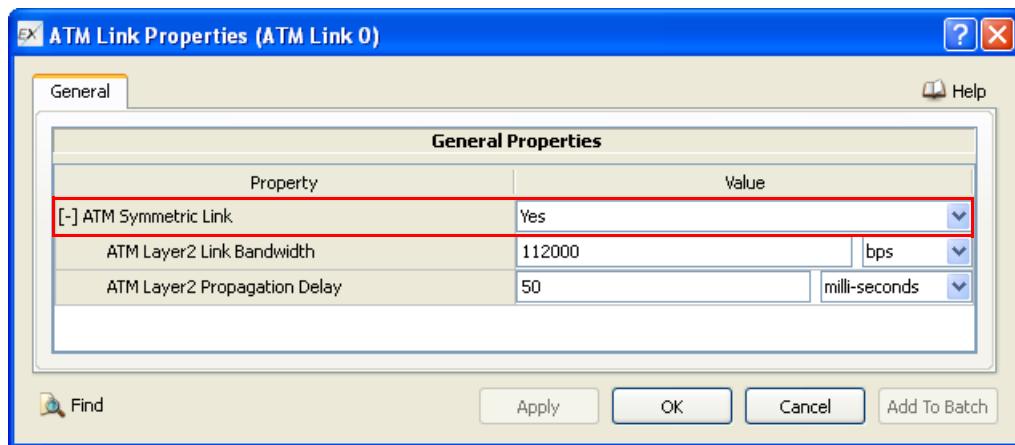


FIGURE 9-6. Setting ATM Link Properties

3. If **ATM Symmetric Link** is set to Yes, set the ATM link parameters listed in [Table 9-2](#).

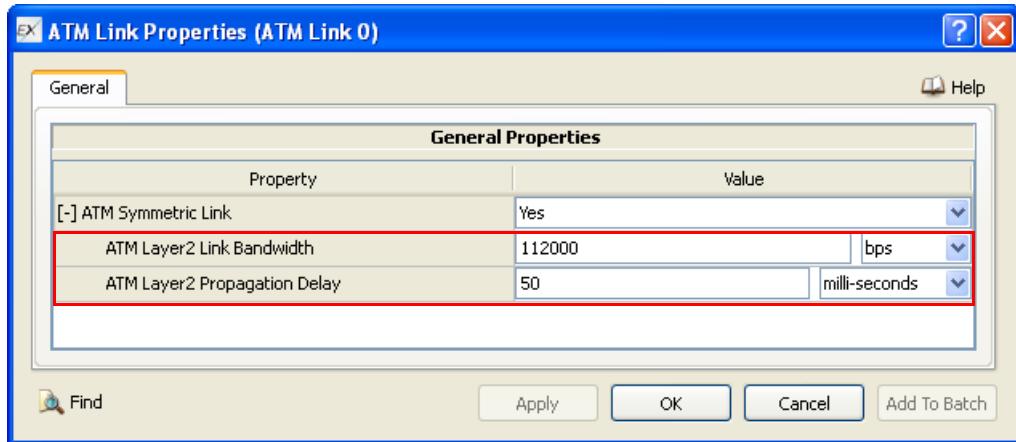


FIGURE 9-7. Setting Symmetric ATM Link Properties

TABLE 9-2. Command Line Equivalent of ATM Link Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|------------------------------|-------------------------|-----------------------------------|
| ATM Layer2 Link Bandwidth | ATM Link, ATM Interface | ATM-LAYER2-LINK-BANDWIDTH |
| ATM Layer2 Propagation Delay | ATM Link, ATM Interface | ATM-LAYER2-LINK-PROPAGATION-DELAY |

9.1.4.2.2 Asymmetric ATM Links

To configure link parameters for an asymmetric ATM link, perform the following steps:

1. Configure the ATM link to be an asymmetric link, as described in [Section 9.1.4.2.1](#).
2. For each interface of the link, configure the ATM link parameters as follows:
 - a. Go to **ATM Interface Properties Editor > ATM Interfaces > ATM Interface # > ATM Layer2**.
 - b. Set the ATM link parameters listed in [Table 9-2](#).

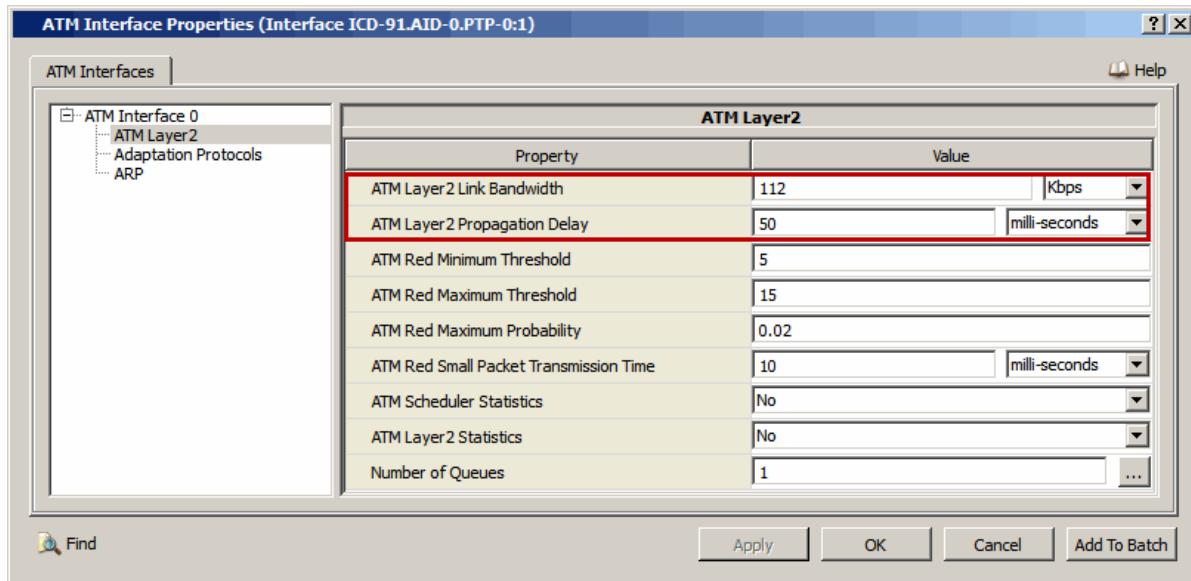


FIGURE 9-8. Setting Asymmetric ATM Link Properties

9.1.4.3 Configuring Adaptation Layer Parameters

To configure ATM Adaptation Layer parameters, perform the following steps:

1. Go to one of the following locations:
 - To set properties for a specific ATM device, go to **ATM Device Properties Editor > Adaptation Protocols**.
 - To set properties for a specific ATM interface, go to one of the following locations:
 - **ATM Device Properties Editor > ATM Interfaces > ATM Interface # > Adaptation Protocols**.
 - **ATM Interface Properties Editor > ATM Interfaces > ATM Interface # > Adaptation Protocols**.

In this section, we show how to configure the ATM Adaptation Layer parameters in the ATM Interface Properties Editor. Parameters can be set in the other properties editors in a similar way.

2. Set **Adaptation Protocol** to *AAL5* and set the dependent parameters listed in [Table 9-3](#).

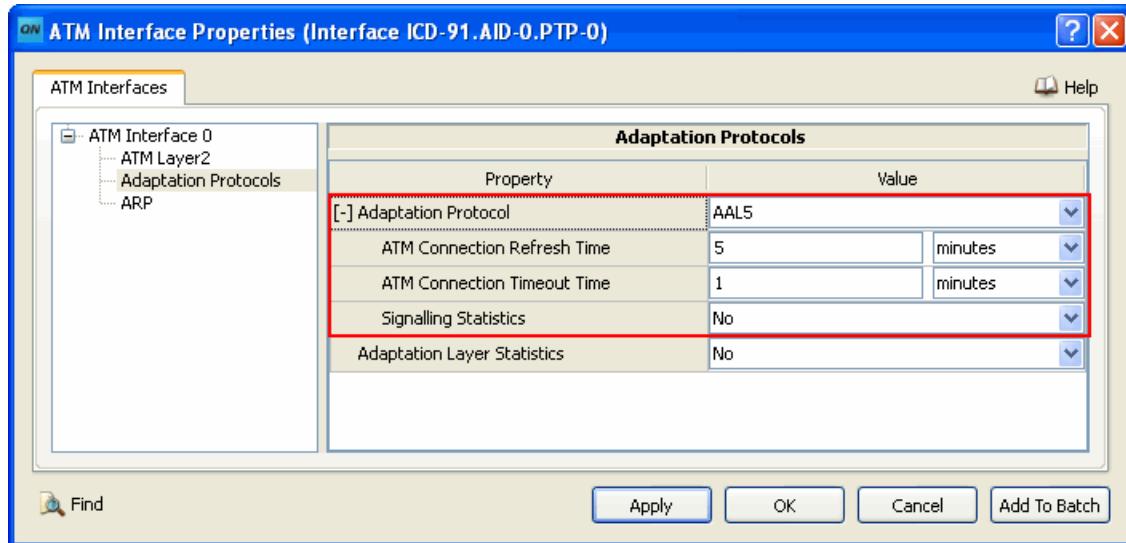


FIGURE 9-9. Setting Adaptation Layer Parameters

TABLE 9-3. Command Line Equivalent of Adaptation Layer Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-----------------------------|---------------------------|-----------------------------|
| Adaptation Protocol | ATM Device, ATM Interface | ADAPTATION-PROTOCOL |
| ATM Connection Refresh Time | ATM Device, ATM Interface | ATM-CONNECTION-REFRESH-TIME |
| ATM Connection Timeout Time | ATM Device, ATM Interface | ATM-CONNECTION-TIMEOUT-TIME |
| Signalling Statistics | ATM Device, ATM Interface | ATM-SIGNALLING-STATISTICS |

3. Set the Adaptation Layer statistics parameters listed in [Table 9-4](#).

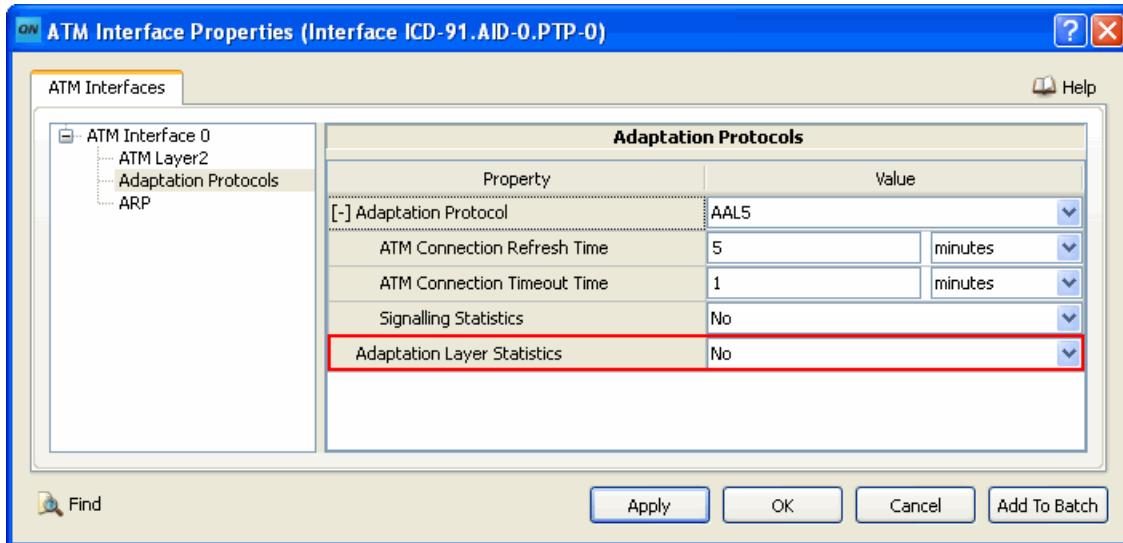


FIGURE 9-10. Setting Adaptation Layer Statistics Parameters

TABLE 9-4. Command Line Equivalent of Adaptation Layer Statistics Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-----------------------------|---------------------------|-----------------------------|
| Adaptation Layer Statistics | ATM Device, ATM Interface | ADAPTATION-LAYER-STATISTICS |

9.1.4.4 Configuring ATM Layer 2 Parameters

To configure ATM Layer 2 parameters, perform the following steps:

1. Go to one of the following locations:

- **ATM Device Properties Editor > ATM Interfaces > ATM Interface # > ATM Layer 2.**
- **ATM Interface Properties Editor > ATM Interfaces > ATM Interface # > ATM Layer 2.**

In this section, we show how to configure the ATM Layer 2 parameters in the ATM Interface Properties Editor. Parameters can be set in the other properties editors in a similar way.

2. Set the ATM Layer 2 parameters listed in [Table 9-5](#).

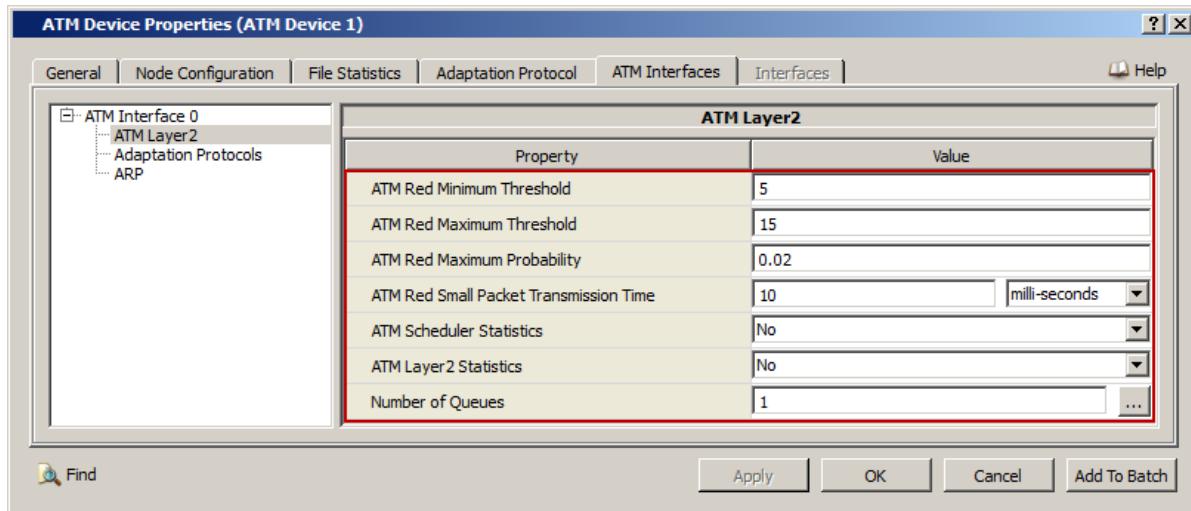


FIGURE 9-11. Setting ATM Layer 2 Parameters

TABLE 9-5. Command Line Equivalent of ATM Layer 2 Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|--|------------------------|--|
| ATM Red Minimum Threshold | ATM Interface | ATM-RED-MIN-THRESHOLD |
| ATM Red Maximum Threshold | ATM Interface | ATM-RED-MAX-THRESHOLD |
| ATM Red Maximum Probability | ATM Interface | ATM-RED-MAX-PROBABILITY |
| ATM Red Small Packet Transmission Time | ATM Interface | ATM-RED-SMALL-PACKET-TRANSMISSION-TIME |
| ATM Scheduler Statistics | ATM Interface | ATM-SCHEDULER-STATISTICS |
| ATM Layer 2 Statistics | ATM Interface | ATM-LAYER2-STATISTICS |
| Number of Queues | ATM Interface | N/A |

Setting Parameters

- Set **Number of Queues** to the number of ATM queues at the interface.
3. To configure the properties for the number of queues, do the following:
- a. Click the **Open Array Editor** button in the **Value** column. This opens the Array Editor ([Figure 9-12](#)).
 - b. Set the ATM queues parameters listed in [Table 9-6](#) for each index.

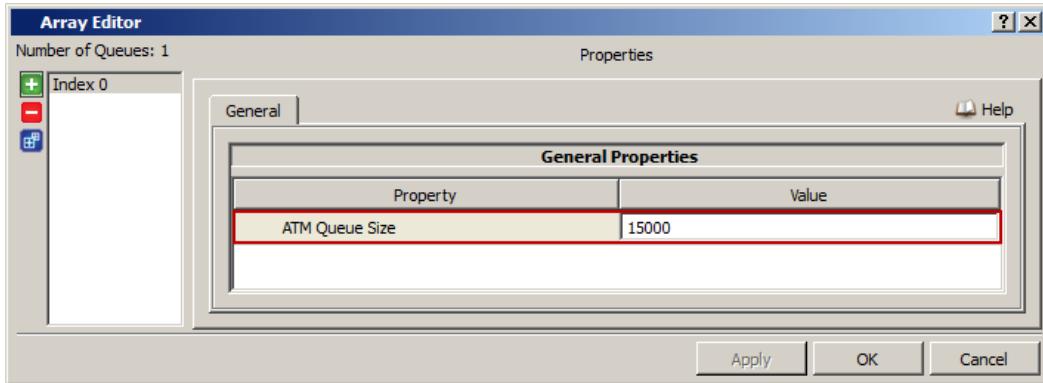


FIGURE 9-12. Setting ATM Layer 2 Parameters

TABLE 9-6. Command Line Equivalent of ATM Layer 2 Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|----------------|------------------------|------------------------|
| ATM Queue Size | ATM Interface | ATM-QUEUE-SIZE |

9.1.4.5 Configuring ATM Static Routes

To configure ATM static routes, perform the following steps:

1. Go to **Scenario Properties Editor > General > ATM Configuration**.
2. Set **Specify ATM Static Routes** to Yes and set the dependent parameters listed in [Table 9-7](#).

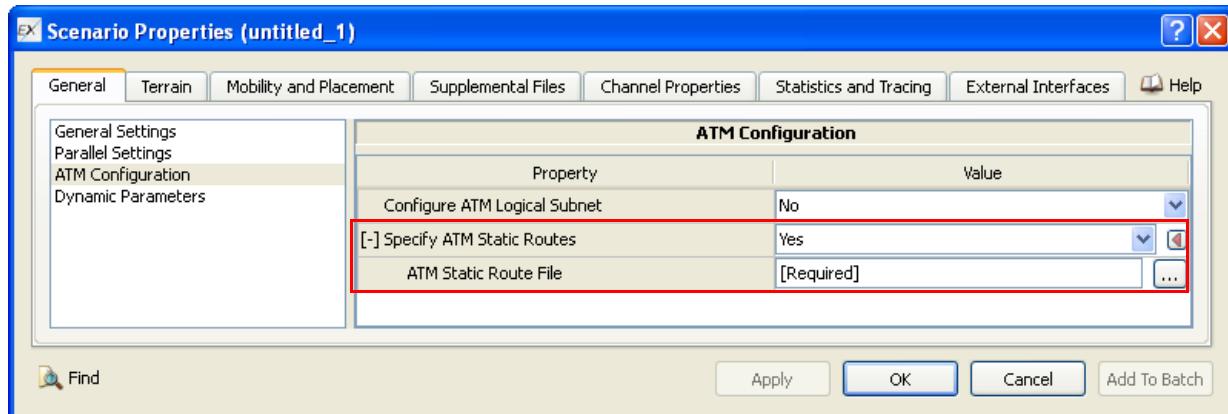


FIGURE 9-13. Setting ATM Static Routes

TABLE 9-7. Specify ATM Static Routes parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-----------------------|------------------------|------------------------|
| ATM Static Route File | Global | ATM-STATIC-ROUTE-FILE |

Setting Parameters

- Set **ATM Static Route File** to the name of the ATM static routes file. The format of the ATM static routes file is described in [Section 9.1.3.1.1](#).

9.1.5 Statistics

Table 9-8 lists the ATM statistics that are output to the statistics (.stat) file at the end of simulation.

TABLE 9-8. ATM Statistics

| Statistic | Description |
|---|--|
| Destination | Next destination, Node connected to the other side of this Point-to-Point link/interface. |
| Cell Received | Total number of cells received in the interface by ATM LAYER2. |
| Cell Forward | Total number of cells forwarded through the interface ATM LAYER2. |
| Cell Discarded for no Route | Total number of cells discarded due to unavailability of route at ATM LAYER2. |
| Control Cell Received | Total number of control cells (signaling cell) received at ATM LAYER2. |
| Cell Delivered to upper layer | Total number of cells delivered to upper layer (adaptation layer) ATM LAYER2. |
| Link Utilization | Link utilization ratio by ATM LAYER2. |
| Number of AAL Service Data Units Sent | Total number of AAL CPCS service data units sent from this node by AAL5. |
| Number of AAL Service Data Units Received | Total number of AAL CPCS service data units received in this node by AAL5. |
| Number of ATM Service Data Units Sent | Total number of SDUs sent to ATM layer 2 after segmentation by AAL5. |
| Number of ATM Service Data Units Received | Total number of SDUs received from ATM layer2 before reassembly by AAL5. |
| Number of ATM Service Data Units Dropped | Total number of SDUs dropped due to error, like intermediate cell drop etc. by AAL5. |
| Number of Packets Segmented | Total number of packets (Data unit) segmented into cells by AAL5. |
| Number of Packets Reassembled | Total number of packets (data units) reassembled from cells by AAL5. |
| Number of SAAL Packets Sent | Total number of signaling packets sent from this node by AAL5. |
| Number of SAAL Packets Received | Total number of signaling packets received in this node by AAL5. |
| Number of Data Packets Sent | Total number of data packets forwarded through this node by AAL5. |
| Number of Data Packets Received | Total number of data packets received in this node by AAL5. |
| Number of Data Packets Received from IP | Total number of data packets received from IP layer by AAL5. |
| Number of Data Packets Forwarded to IP | Total number of data packets forwarded to IP layer by AAL5. |
| Number of IP Packets Discarded | Total number of erroneous data packets (multicast/control packet) discarded. These packets are not further processed within ATM cloud by AAL5. |
| Number of IP Packets Dropped | Total number of data packets dropped due to no route after processing within ATM cloud by AAL5. |
| Number of Alert Messages Sent | Total number of alert messages sent by SAAL. |
| Number of Alert Messages Received | Total number of alert messages received by SAAL. |
| Number of Connect Messages Sent | Total number of connect messages sent by SAAL. |
| Number of Connect Messages Received | Total number of connect messages received by SAAL. |
| Number of Setup Messages Sent | Total number of setup messages sent by SAAL. |

TABLE 9-8. ATM Statistics (Continued)

| Statistic | Description |
|-----------------------------------|---|
| Number of Setup Messages Received | Total number of setup messages received by SAAL. |
| Number of Packets Queued | Total number of packets enqueued. |
| Number of Packets Dequeued | Total number of packets dequeued |
| Number of Packets Dropped | Total number of packets dropped |
| Service Ratio | Ratio of total number of packets dequeued to total number of dequeue requests received. |

9.1.6 Sample Scenarios

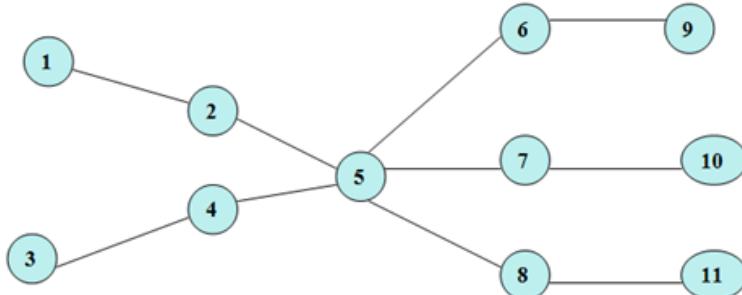
9.1.6.1 Standalone ATM Backbone Scenario

9.1.6.1.1 Scenario Description

The scenario in [Figure 9-14](#) shows an ATM scenario. Nodes 1 to 11 are ATM nodes and they are connected by ATM links as shown in the figure. Nodes 1, 3, 9, 10, 11 are ATM end systems. Nodes 2, 4, 5, 6, 7, 8 are ATM switches. Adaptation protocol is set to AAL5 and ATM static route is configured.

Topology

The scenario topology is shown in [Figure 9-14](#).

**FIGURE 9-14. Standalone ATM Network Scenario**

9.1.6.1.2 Command Line Configuration

Set the parameters in the configuration file as described in the following steps.

1. Configure the ATM links:

```
ATM-LAYER2-LINK-BANDWIDTH      112000
ATM-LAYER2-LINK-PROPAGATION-DELAY   10MS

ATM-LINK ICD-91.AID-0.PTP-0 { 1, 2 }
ATM-LINK ICD-91.AID-0.PTP-1 { 2, 5 }
ATM-LINK ICD-91.AID-0.PTP-2 { 3, 4 }
ATM-LINK ICD-91.AID-0.PTP-3 { 4, 5 }
ATM-LINK ICD-91.AID-0.PTP-4 { 5, 6 }
ATM-LINK ICD-91.AID-0.PTP-5 { 5, 7 }
ATM-LINK ICD-91.AID-0.PTP-6 { 5, 8 }
ATM-LINK ICD-91.AID-0.PTP-7 { 6, 9 }
ATM-LINK ICD-91.AID-0.PTP-8 { 7, 10 }
ATM-LINK ICD-91.AID-0.PTP-9 { 8, 11 }

[1 thru 11] ATM-NODE YES
[1 3 9 10 11] ATM-END-SYSTEM YES
```

2. Configure Adaptation layer parameters.

```
ADAPTATION-PROTOCOL AAL5

ATM-RED-MIN-THRESHOLD      5
ATM-RED-MAX-THRESHOLD      15
ATM-RED-MAX-PROBABILITY    0.02
ATM-RED-SMALL-PACKET-TRANSMISSION-TIME   10MS
```

3. Configure the logical subnet.

The end system nodes should be placed in a logical subnet defined as the following:

```
ATM-LOGICAL-SUBNET N8-192.168.5.0 {1, 3, 9, 10, 11}
```

4. Configure ATM static routes.

ATM routes packets using static routes.

```
ATM-STATIC-ROUTE YES
ATM-STATIC-ROUTE-FILE   ./atm.routes-static
```

For this sample scenario, the content of the static route file is given as below.

Note: Only routes to end system nodes are specified as the source, and destination nodes must be end system nodes.

```
1 ICD-91.AID-0.PTP-0:1 0
1 ICD-91.AID-0.PTP-2:3 0
1 ICD-91.AID-0.PTP-7:9 0
1 ICD-91.AID-0.PTP-8:10 0
1 ICD-91.AID-0.PTP-9:11 0

2 ICD-91.AID-0.PTP-0:1 0
2 ICD-91.AID-0.PTP-2:3 1
2 ICD-91.AID-0.PTP-7:9 1
2 ICD-91.AID-0.PTP-8:10 1
2 ICD-91.AID-0.PTP-9:11 1

3 ICD-91.AID-0.PTP-0:1 0
3 ICD-91.AID-0.PTP-2:3 0
3 ICD-91.AID-0.PTP-7:9 0
3 ICD-91.AID-0.PTP-8:10 0
3 ICD-91.AID-0.PTP-9:11 0

4 ICD-91.AID-0.PTP-0:1 1
4 ICD-91.AID-0.PTP-2:3 0
4 ICD-91.AID-0.PTP-7:9 1
4 ICD-91.AID-0.PTP-8:10 1
4 ICD-91.AID-0.PTP-9:11 1

5 ICD-91.AID-0.PTP-0:1 0
5 ICD-91.AID-0.PTP-2:3 1
5 ICD-91.AID-0.PTP-7:9 2
5 ICD-91.AID-0.PTP-8:10 3
5 ICD-91.AID-0.PTP-9:11 4

6 ICD-91.AID-0.PTP-0:1 0
6 ICD-91.AID-0.PTP-2:3 0
6 ICD-91.AID-0.PTP-7:9 1
6 ICD-91.AID-0.PTP-8:10 0
6 ICD-91.AID-0.PTP-9:11 0

7 ICD-91.AID-0.PTP-0:1 0
7 ICD-91.AID-0.PTP-2:3 0
7 ICD-91.AID-0.PTP-7:9 0
7 ICD-91.AID-0.PTP-8:10 1
7 ICD-91.AID-0.PTP-9:11 0

8 ICD-91.AID-0.PTP-0:1 0
8 ICD-91.AID-0.PTP-2:3 0
8 ICD-91.AID-0.PTP-7:9 0
8 ICD-91.AID-0.PTP-8:10 0
8 ICD-91.AID-0.PTP-9:11 1
```

```

9 ICD-91.AID-0.PTP-0:1 0
9 ICD-91.AID-0.PTP-2:3 0
9 ICD-91.AID-0.PTP-7:9 0
9 ICD-91.AID-0.PTP-8:10 0
9 ICD-91.AID-0.PTP-9:11 0

10 ICD-91.AID-0.PTP-0:1 0
10 ICD-91.AID-0.PTP-2:3 0
10 ICD-91.AID-0.PTP-7:9 0
10 ICD-91.AID-0.PTP-8:10 0
10 ICD-91.AID-0.PTP-9:11 0

11 ICD-91.AID-0.PTP-0:1 0
11 ICD-91.AID-0.PTP-2:3 0
11 ICD-91.AID-0.PTP-7:9 0
11 ICD-91.AID-0.PTP-8:10 0
11 ICD-91.AID-0.PTP-9:11 0

```

5. Configure application.

Only CBR applications are supported in a standalone ATM networks. Source and destination nodes must be ATM end system nodes. For example, a sample application file defined in *.app file, is as follows:

```

CBR 1 11 1 512 1M 1M 30M
CBR 11 1 1 512 1M 1M 30M
CBR 3 9 1 512 1M 1M 30M
CBR 9 3 1 512 1M 1M 30M

```

To transmit the packets, paths are established dynamically using SVC (for example, the translation table entry is created through the exchange of various signaling messages). Once the path is recognized, cells are transmitted through it.

6. Enable statistics collection.

| | |
|-----------------------------|-----|
| ADAPTATION-LAYER-STATISTICS | YES |
| ATM-SIGNALLING-STATISTICS | YES |
| ATM-SCHEDULER-STATISTICS | YES |
| ATM-QUEUE-STATISTICS | YES |
| ATM-LAYER2-STATISTICS | YES |

9.1.6.1.3 GUI Configuration

Follows these steps to configure ATM using the GUI:

1. Create a new scenario. Place 11 ATM nodes as shown in topology.
2. For nodes 1, 3, 9, 10, 11, set **Node Type** to *ATM End System*, as shown in [Figure 9-3](#).
3. For all Links, set **ATM Layer2 Link Bandwidth** to *112000 bps* and **ATM Layer2 Link Propagation Delay** as *50MS* as shown in [Figure 9-7](#).
4. For all nodes, set the Adaptation Layer parameters, as shown in [Figure 9-9](#) and [Figure 9-10](#).
5. For all interfaces, set the ATM Layer 2 parameters, as shown in [Figure 9-11](#).

9.1.6.2 IP over ATM Scenario

9.1.6.2.1 Scenario Description

This sample is an IP over ATM scenario consisting of 12 nodes, among which 3, 4, 5, 6, 7 and 8 are present in ATM cloud and the rest are in IP cloud. As node 3, 7 and 8 have interface both in IP and ATM, thus they are considered as end-systems.

Topology

Figure 9-15 shows the topology of the IP over ATM scenario.

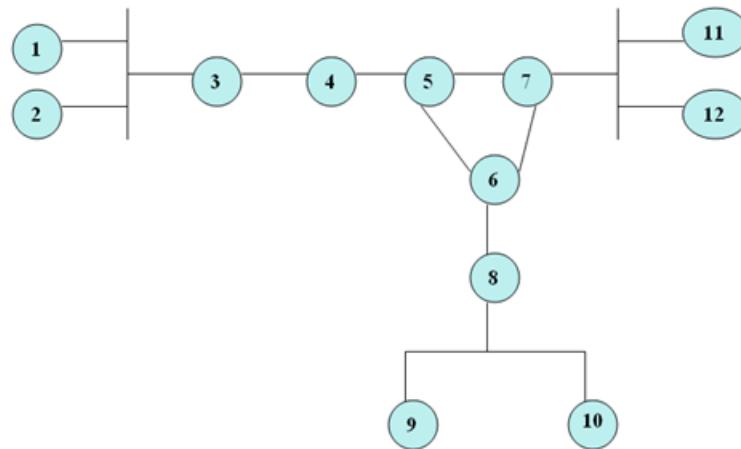


FIGURE 9-15. IP Over ATM Scenario

9.1.6.2.2 Command Line Configuration

Scenario Configuration (.config) File Parameters

Set the parameters in the scenario configuration file as described in the following steps.

1. Configure the ATM backbone:

```
ATM-LAYER2-LINK-BANDWIDTH      1112000
ATM-LAYER2-LINK-PROPAGATION-DELAY 10MS
```

2. Configure the ATM cloud for nodes 3, 4, 5, 6, 7 and 8.

```
ATM-LINK ICD-91.AID-1.PTP-1 {3, 4}
ATM-LINK ICD-91.AID-1.PTP-2 {4, 5}
ATM-LINK ICD-91.AID-1.PTP-3 {5, 6}
ATM-LINK ICD-91.AID-1.PTP-4 {6, 8}
ATM-LINK ICD-91.AID-1.PTP-5 {6, 7}
```

```
[3 4 5 6 7 8] ATM-NODE YES
[3 7 8] ATM-END-SYSTEM YES
```

3. Configure the IP network.

```
SUBNET N8-1.0 {1 thru 3}

SUBNET N8-2.0 {7, 11, 12}

SUBNET N8-3.0 {8 thru 10}

LINK-BANDWIDTH           112000
LINK-PROPAGATION-DELAY    50MS
```

4. Configure default gateways.

```
[1 2 3] DEFAULT-GATEWAY 3
[7 11 12] DEFAULT-GATEWAY 7
[8 9 10] DEFAULT-GATEWAY 8
```

5. Configure the logical subnet.

```
ATM-LOGICAL-SUBNET N8-8.0 {3, 7, 8}

ADAPTATION-PROTOCOL AAL5
ADAPTATION-LAYER-STATISTICS YES
ATM-STATIC-ROUTE YES
ATM-STATIC-ROUTE-FILE atm.route-static
ARP-ENABLED YES
ARP-USE-BUFFER YES
ARP-STATIC-CACHE-FILE atm.arp-static
ARP-STATISTICS YES

ATM-SIGNALLING-STATISTICS      YES
ATM-SCHEDULER-STATISTICS       NO
ATM-LAYER2-STATISTICS          YES
```

6. Configure optional parameters.

```
ATM-CONNECTION-REFRESH-TIME   15M
ATM-CONNECTION-TIMEOUT-TIME   2M

ATM-RED-MIN-THRESHOLD         5
ATM-RED-MAX-THRESHOLD         15
ATM-RED-MAX-PROBABILITY       0.02
ATM-RED-SMALL-PACKET-TRANSMISSION-TIME 10MS

NETWORK-PROTOCOL IP
MAC-PROTOCOL MAC802.3
ROUTING-PROTOCOL OSPFv2
```

ARP Static Cache File

Create the ARP static cache file (atm.arp_static). The contents of the file are:

```
3 IP 0.0.8.1 ATM 23:4f:5C:aa:FE:B2 0
7 IP 0.0.8.2 ATM 5C-AA-FE-23-4F-C2 0
8 IP 0.0.8.3 ATM 5C-BA-FE-23-4F-F2 0
```

Application Configuration File

Configure a CBR session from node 1 to node 12 by entering the following line the application configuration (atm.app) file:

```
CBR 1 12 0 512 1M 20S 0M
```

9.1.6.2.3 GUI Configuration

To configure the scenario in QualNet GUI, perform the steps described below. Note that you must select the ATM button in the Devices tab to create ATM nodes.

1. Create a new scenario. Place 6 ATM nodes as shown in topology.
2. For nodes 3, 7, and 8, set **Node Type** to *ATM End System*, as shown in [Figure 9-3](#).
3. For all links, set **ATM Layer2 Link Bandwidth** to *112000 bps* and **ATM Layer2 Link Propagation Delay** to *50MS*, as shown in [Figure 9-7](#).
4. For nodes 3, 4,5,6,7 and 8, set the Adaptation Layer parameters, as shown in [Figure 9-9](#).
5. Set **ATM Logical Subnet** to *Yes*, as shown in [Figure 9-4](#).
6. Set **Specify ATM Static Routes** to *Yes* and specify the static routes file, as shown in [Figure 9-13](#).

7. Node 1 is a non-ATM node and only has an IP Interface. To enable node 1 to communicate with an ATM network, do the following:
- Go to **Default Device Properties Editor > Node Configuration > Routing Protocols**.
 - Set **Configure Default Gateway** to Yes and set **Default Gateway** to the gateway's node ID or IP address.

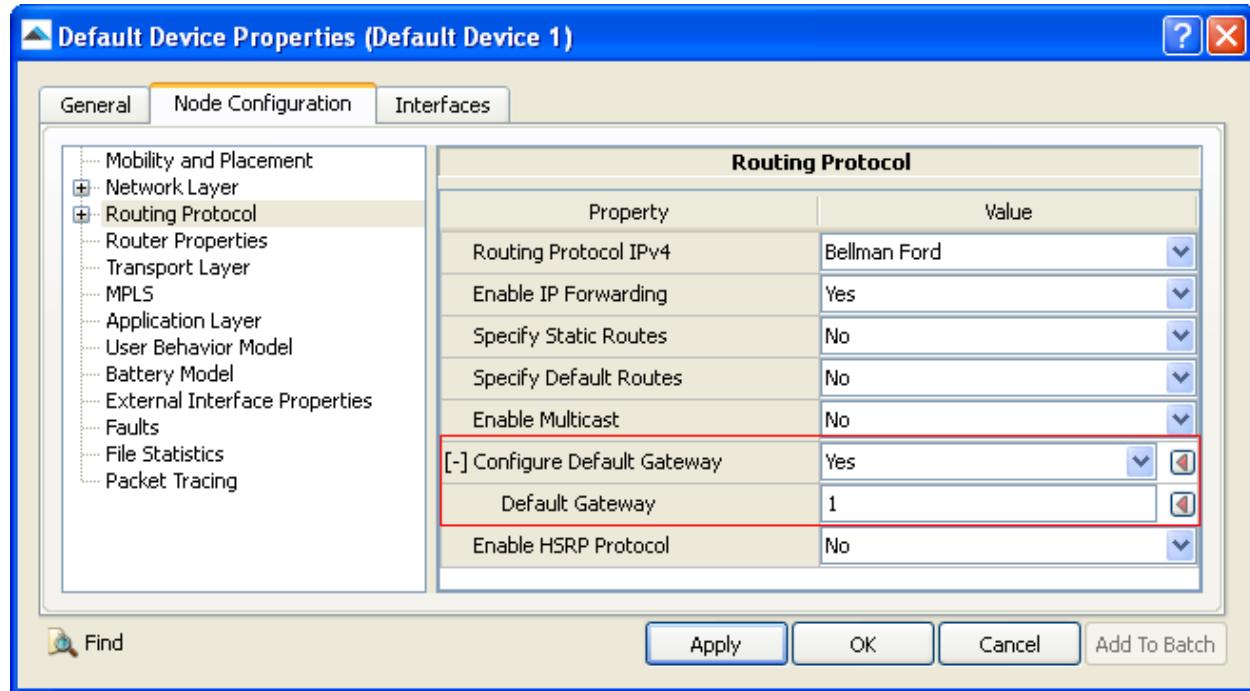


FIGURE 9-16. Setting Default Gateway

8. For all interfaces, set the ATM Layer 2 parameters, as shown in [Figure 9-11](#).

9.1.7 Scenarios Included in QualNet

The QualNet distribution includes several sample scenarios for the ATM model. All scenarios are located in the directory `QUALNET_HOME/scenarios/developer`. [Table 9-9](#) lists the sub-directory where each scenario is located.

TABLE 9-9. ATM Scenarios Included in QualNet

| Scenario | Description |
|--|--|
| <code>atm/bandwidth-checking/less-bw</code> | Shows bandwidth utilization and if requested bandwidth exceeds the limit, then whether bandwidth is properly refused or not. |
| <code>atm/bandwidth-checking/release-bw</code> | Shows whether the bandwidth is released as soon as the application terminates. |
| <code>atm/erroneous/src-not-es</code> | Shows the case where the source is not an end-system. |
| <code>atm/general/duplex-app</code> | Shows general applications with four duplex CBR applications. |
| <code>atm/general/simplex-app</code> | Shows general applications with four simplex CBR applications. |
| <code>atm/multiplexing/queue-check</code> | Shows asynchronous multiplexing. |

TABLE 9-9. ATM Scenarios Included in QualNet (Continued)

| Scenario | Description |
|-------------------------|--|
| atm-ip/addrChkScn | Shows whether ATM address is generated for each node. |
| atm-ip/atm-ip-app | Shows whether UDP and TCP based application is working properly |
| atm-ip/atm-ip-wireless | Shows whether ATM-IP is working properly when IP-cloud is a wireless-network. |
| atm-ip/bwCheckScn | Shows whether bandwidth is properly allocated and freed accordingly for each application. |
| atm-ip/failure-ip-cloud | Shows whether ATM-IP is working properly when interface failure occurs inside IP-cloud. |
| atm-ip/saalChkScn | Shows whether signaling messages are exchanged properly and path is discovered. |
| atm-ip/Traffic-Gen | Shows traffic generated from source node for destination node situated in the same IP-cloud, as well as when it's situated in another IP-cloud, separated by ATM-backbone. |

9.1.8 References

1. RFC 2225 "Classical IP and ARP over ATM" M. Laubach, J. Halpern. April 1998.
2. RFC 2684 "Multiprotocol Encapsulation over ATM Adaptation Layer 5" D. Grossman, J. Heinanen. September 1999
3. ATM Forum Addressing Specification: Reference Guide AF-RA-0106.000

10 Interfaces

This chapter describes features, configuration requirements and parameters, statistics, and scenarios for External Interfaces in the Developer Model Library, and consists of the following sections:

- AGI Satellite Toolkit (STK) Interface

10.1 AGI Satellite Toolkit (STK) Interface

The AGI Satellite Toolkit (STK) Interface replaces the STK/Connect interface of the earlier QualNet releases.

10.1.1 Description

The STK interface provides a way to interface QualNet with the Satellite Toolkit (STK) developed by Analytical Graphics, Inc. (AGI). It allows the QualNet simulation to use STK's wireless propagation module and antenna models for signal propagation. In addition, it allows STK to provide node position updates, thus controlling the mobility.

10.1.2 Features and Assumptions

This section describes the implemented features, omitted features, assumptions and limitations of the QualNet STK interface.

10.1.2.1 Implemented Features

- Communicate with STK.
- Use STK's wireless propagation module for signal propagation, replacing QualNet's propagation module.
- Use STK's antenna model for antenna related calculations.
- Allow STK to control node mobility.

10.1.2.2 Omitted Features

None.

10.1.2.3 Assumptions and Limitations

When running a QualNet simulation scenario with the STK interface, the corresponding STK scenario files are required by STK. It is very hard, if not impossible, to manually create STK scenario files corresponding to a QualNet simulation scenario. To use QualNet with STK, the scenario should be created in STK first and the corresponding QualNet scenario should be created using the QualNet Interface module of STK (part of STK 9.2 or later releases).

10.1.3 Supplemental Information

The STK interface is supported only on Windows platforms.

10.1.4 Integrating QualNet and STK

The AGI STK Interface is part of the Developer Mode Library. It is enabled by default. No recompilation is needed. However, if you wish to disable it or want to re-enable it, follow the steps below.

Note: Compiling the STK interface doesn't require STK to be installed. But STK 9.2 (or later) with its QualNet Interface Module must be installed to run a QualNet simulation with the STK interface.

To enable or disable the STK interface, follow these steps:

1. To enable the STK interface, add the following to `QUALNET_HOME/main/Makefile-addons-windows` after the `INSERT INTERFACES HERE` comment:

```
include ../../interfaces/agl/Makefile-windows
```

To disable the STK interface, comment out or delete the above line.

2. Make a copy of the makefile for your system. For example, for VC9 on a 32-bit platform, use the following command to make a copy of the makefile:

```
copy Makefile-windows-vc9 Makefile
```

Refer to *QualNet Programmer's Guide* for a list of makefiles for other platforms.

3. Use the following commands to remove all object (.obj) files and recompile QualNet:

```
nmake clean  
nmake
```

Note: Installation of STK 9.2 or later release with its QualNet Interface Module is required for runtime.

10.1.5 Command Line Configuration

To enable QualNet to communicate with STK, include the following parameter in the scenario configuration (.config) file:

```
AGI - INTERFACE YES
```

Note: If the QualNet scenario is created using the QualNet Interface module of STK (version 9.2 or later), the parameter `AGI - INTERFACE` is automatically included in the scenario configuration (.config) file and is set to `YES`. The other STK interface parameters (see [Table 10-1](#)) are also automatically included in the scenario configuration file and are set to their default values.

Configuration Requirements

The Latitude-Longitude-Altitude coordinate system must be used when communicating with STK, i.e., `COORDINATE - SYSTEM` must be set to `LATLONALT`.

STK Interface Parameters

[Table 10-1](#) shows the parameters for the STK interface. See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 10-1. STK Interface Parameters

| Parameter | Value | Description |
|--|----------------------------|---|
| AGI - POSITION - UPDATE - INTERVAL <i>Optional</i> Scope: Global, Node | Time <i>Default: 1S</i> | Specifies how frequently QualNet queries STK for node positions and updates the node positions in the QualNet GUI during execution. |

10.1.6 GUI Configuration

In QualNet GUI, STK related parameters can be configured at scenario level. In addition, some parameters can be configured at node level.

General Configuration

To configure the general STK parameters, perform the following steps:

1. Go to one of the following locations:
 - To set properties at the scenario level, go to **Scenario Properties > External Interfaces**.
 - To set properties at the node level, go to **Default Device Properties Editor > Node Configuration > External Interface Properties > STK Interface**.
- In this section we show how to configure global STK parameters using the Scenario Properties Editor. Parameters can be set in the Default Device Properties Editor in a similar way.
2. Set **Enable AGI Interface** to Yes and set the dependent parameters listed in [Table 10-1](#).

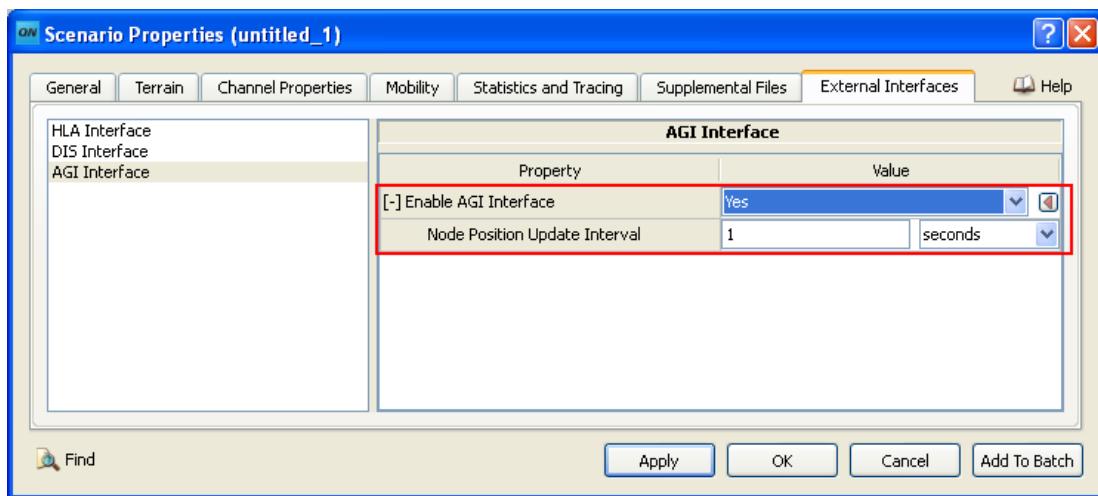


FIGURE 10-1. Setting AGI STK Interface Parameters

TABLE 10-2. Command Line Equivalent of AGI STK Interface Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|-------------------------------|------------------------|------------------------------|
| Node Position Update Interval | Global, Node | AGI-POSITION-UPDATE-INTERVAL |

10.1.7 Statistics

There are no statistics collected for the STK interface.

11 Miscellaneous Models

This chapter describes features, configuration requirements and parameters, statistics, and scenarios for Miscellaneous Models, and consists of the following sections:

- Faults
- File-based Node Placement Model
- Grid Node Placement Model
- Random Node Placement Model
- Uniform Node Placement Model

11.1 Faults

11.1.1 Description

Interface faults are of two types: static and dynamic. A static fault occurs only once and has precise start and end times specified by the user. A dynamic fault can occur repeatedly: the time between successive occurrences (Mean Time Before Failure) and the duration of each occurrence (Mean Repair Time) are specified as probability distributions.

11.1.2 Command Line Configuration

Faults are described by in a Fault Configuration file. To specify the name of the Fault Configuration file, include the following parameter in the scenario configuration (.config) file:

```
FAULT-CONFIG-FILE      <fault-config-file>
```

where

<fault-config-file> Name of the Fault Configuration file.

The format of the Fault Configuration file is described in [Section 11.1.2.1](#).

11.1.2.1 Format of the Fault Configuration

Static and dynamic faults are described in the Fault Configuration file using the syntax described in [Section 11.1.2.1.1](#) and [Section 11.1.2.1.2](#)

11.1.2.1.1 Static Faults

To specify a static fault, include the following statement in the Fault Configuration (.fault) file:

```
INTERFACE-FAULT <interface-address> <start-time> <end-time>
                [INTERFACE-CARD-FAULT <interface-MAC-address>]
```

Note: All parameters should be entered on the same line.

The static fault parameters are described in [Table 11-1](#). See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 11-1. Static Fault Parameters

| Parameter | Value | Description |
|--|---------------------------|---|
| <interface-address> <i>Required</i> | IP Address | Interface address on which the fault is configured. |
| <start-time> <i>Required</i> | Time <i>Range: ≥ 0</i> | Time when the fault starts. |

TABLE 11-1. Static Fault Parameters (Continued)

| Parameter | Value | Description |
|--|--------------------------------|--|
| <end-time> <i>Required</i> | Time <i>Range:</i> ≥ 0 | Time when the fault ends. |
| INTERFACE-CARD-FAULT <interface-mac-address> <i>Optional</i> | String | MAC address of the network interface card after the faulty link comes up again. The value of this parameter must be a MAC address. Note: Replacement MAC address is supported only if ARP is enabled. |

- Notes:**
1. The value of <end-time> should be 0 or greater than or equal to the value of <start-time>.
 2. If <end-time> is set to 0, the fault continues until the end of simulation.

Examples of Parameter Usage

The following are examples of static fault configuration:

1. The interface with IP address 192.0.0.2 will be down from time 0 second to 10 seconds. After that, the MAC address of the network interface card of this interface becomes 5C-AA-FE-23-4F-C2 when the link comes up again after 10 seconds.

```
INTERFACE-FAULT 192.0.0.2 0S 10S INTERFACE-CARD-FAULT 5C-AA-FE-23-4F-C2
```

2. Link faults can be modeled using two INTERFACE-FAULT statements to specify faults on both interfaces associated with the link. For example, to specify a link fault for link N8-192.0.1.0, use:

```
INTERFACE-FAULT 192.0.1.1 0S 10S
INTERFACE-FAULT 192.0.1.2 0S 10S
```

This means that the link between interface addresses 192.0.1.1 and 192.0.1.2 will be down from 0 second to 10 seconds.

11.1.2.1.2 Dynamic Faults

To specify a dynamic fault, include the following statement in the Fault Configuration (.fault) file:

```
INTERFACE-FAULT <interface-address> REPS <num-repetitions>
                      START <start-time> MTBF <time-between-failures>
                           DOWN <repair-time>
```

Note: All parameters should be entered on the same line.

The dynamic fault parameters are described in [Table 11-2](#).

TABLE 11-2. Dynamic Fault Parameters

| Parameter | Value | Description |
|---|---|--|
| <interface-address> <i>Required</i> | IP Address | Interface address on which the fault is configured. |
| REPS <num-repetitions> <i>Required</i> | Integer distribution (see note below this table) | <p>Number of times the interface will fail.</p> <p>A value of 0 means failures occur continually until the end of simulation. A number greater than 0 indicates a specific number of failures.</p> <p>The number of repetitions is specified as an integer distribution. The value 0 can be specified in the following three ways:</p> <ul style="list-style-type: none"> • DET 0 • UNI 0 0 • EXP 0 |
| START <start-time> <i>Required</i> | Time distribution (see note below this table) | Time before the first failure. |
| MTBF <time-between-failures> <i>Required</i> | Time distribution (see note below this table) | Length of time between successive failures. |
| DOWN <repair-time> <i>Required</i> | Time distribution (see note below this table) | Duration of a failure. i.e., the length of time for which the interface is down. |

Note: Integer Distributions and Time Distributions: The number of failures, start time, mean time between failures, and down time are specified as random number distributions. Three random number distributions are supported: deterministic, uniform, and exponential.

- Deterministic distribution: This is equivalent to a constant value. This distribution is specified as:

```
DET <value>
```

It always returns <value> as the value.

- Uniform distribution: This distribution is specified as:

```
UNI <value-1> <value-2>
```

It returns a value uniformly distributed between <value-1> and <value-2>.

- Exponential distribution: This distribution is specified as:

```
EXP <value>
```

It returns a value from an exponential distribution with <value> as the mean.

For integer distributions, <value>, <value-1>, and <value-2> are integer values, e.g., 0, 10, 15, etc.

For time distributions, <value>, <value-1>, and <value-2> are time values, e.g., 5S, 0.5MS, 100US, etc.

Examples of Parameter Usage

The following are examples of dynamic fault configuration:

1. The interface with IP address 192.0.1.1 will fail once 1 second after the start of the simulation for a length of time taken from an exponential distribution with 1 second as the mean.

```
INTERFACE-FAULT 192.0.1.1 REPS DET 1 START DET 1S MTBF UNI 1S 2S DOWN  
EXP 1S
```

2. The interface with IP address 192.0.0.2 will fail three times, each time for a period of 10 seconds. The first failure will occur 15 seconds after the start of simulation. The time between successive failures would be a uniformly distributed value between 5 seconds and 10 seconds.

```
INTERFACE-FAULT 192.0.0.2 REPS DET 3 START DET 15S MTBF UNI 5S 10S DOWN  
DET 10S
```

11.1.3 GUI Configuration

General Configuration

To configure faults in the GUI, perform the following steps:

1. Go to one of the following locations:
 - To configure faults for a specific point-to-point link, go to **Point-to-point Link Properties Editor > Point-to-point Link Properties > Faults**.
 - To configure faults for a specific node, go to **Default Device Properties Editor > Node Configuration > Faults**.
 - To configure faults for a specific interface of a node, go to one of the following locations:
 - **Interface Properties Editor > Interfaces > Interface # > Faults**
 - **Default Device Properties Editor > Interfaces > Interface # > Faults**.

In this section, we show how to configure faults parameters for a specific node using the Default Device Properties Editor. Faults can be configured in the other properties editors in a similar way.

2. Go to **Default Device Properties Editor > Node Configuration > Faults**. The Faults Editor is displayed in the right panel.



FIGURE 11-1. Configuring Faults for a Node

3. To add a fault, click on the  button. This adds a fault to the list. To remove a fault, select the fault from the list and click on the  button

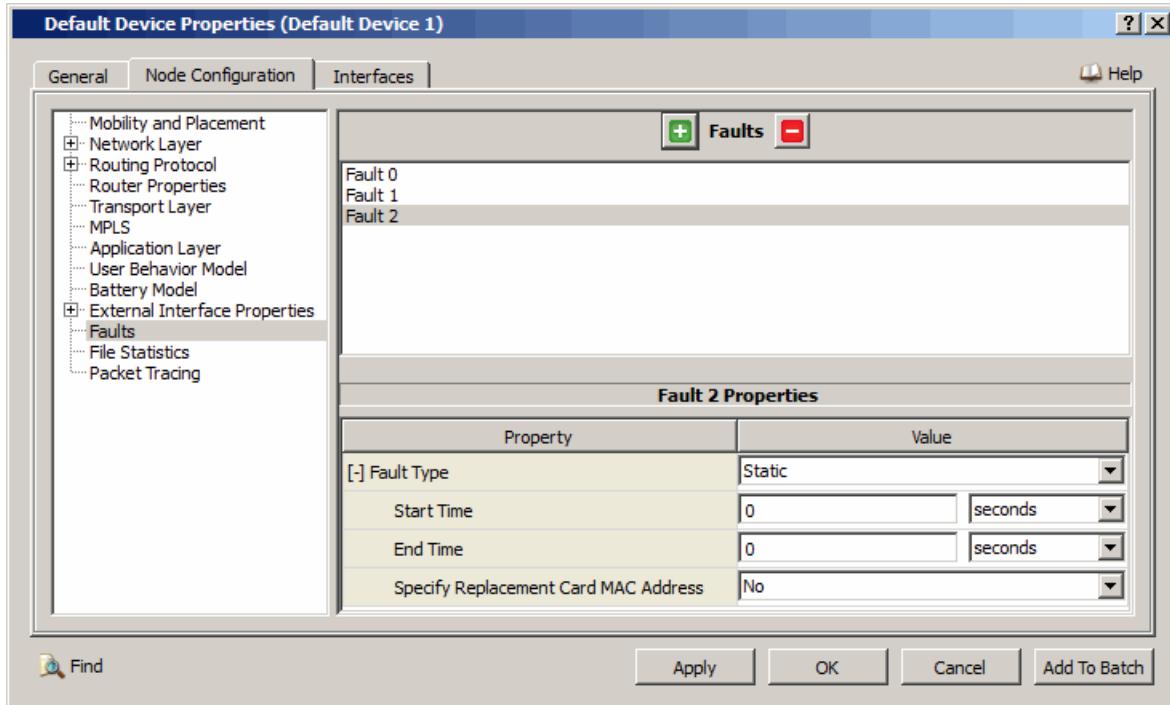


FIGURE 11-2. Adding and Deleting Faults

4. To configure the properties of a fault, select the fault from the list and edit its properties in the editor that opens in the bottom part of the panel.

Configuring Static Faults

To configure a static fault, do the following:

1. Set **Fault Type** to *Static*, and set the dependent parameters listed in [Table 11-3](#).

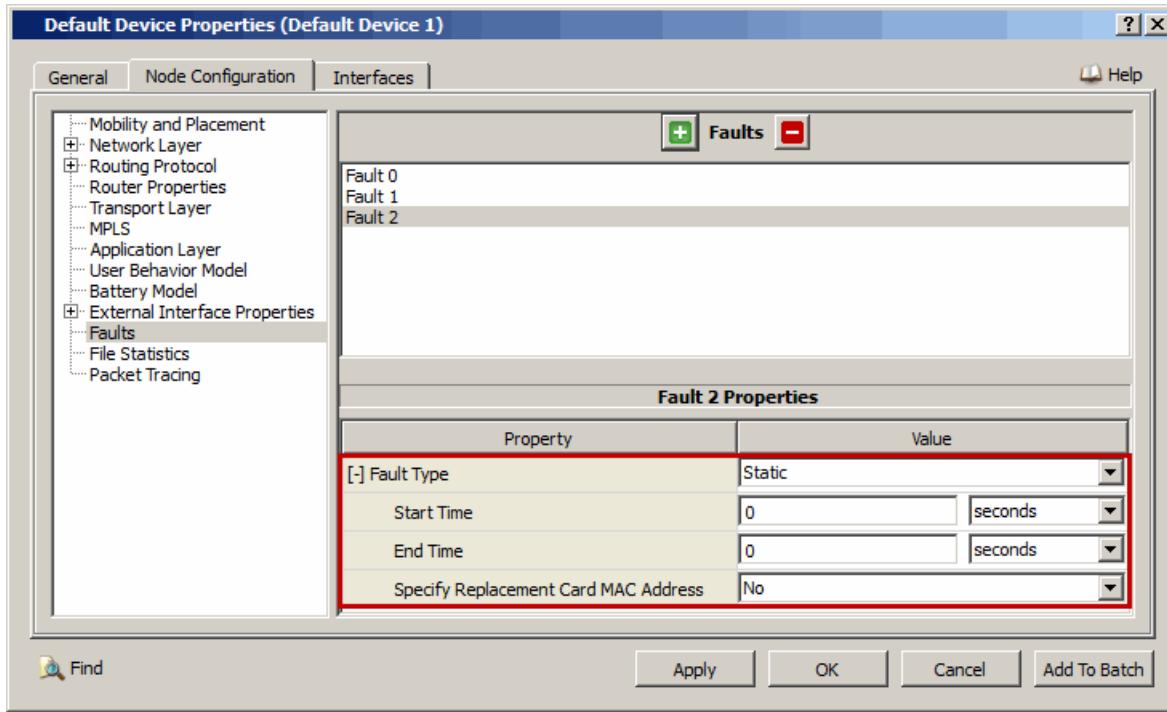


FIGURE 11-3. Setting Static Fault Parameters

TABLE 11-3. Command Line Equivalent of Static Fault Type Parameters

| GUI Parameter | Command Line Parameter |
|---|------------------------|
| Start Time | <start-time> |
| End Time | <end-time> |
| Specify Replacement Card MAC Address (set to Yes) | INTERFACE-CARD-FAULT |

Setting Parameters

- To specify a replacement MAC address for the fault, set **Specify Replacement Card MAC Address** to Yes.

Note: Replacement MAC address is supported only if ARP is enabled.

2. If **Specify Replacement Card MAC Address** is set to Yes, then set the dependent parameters listed in Table 11-4.

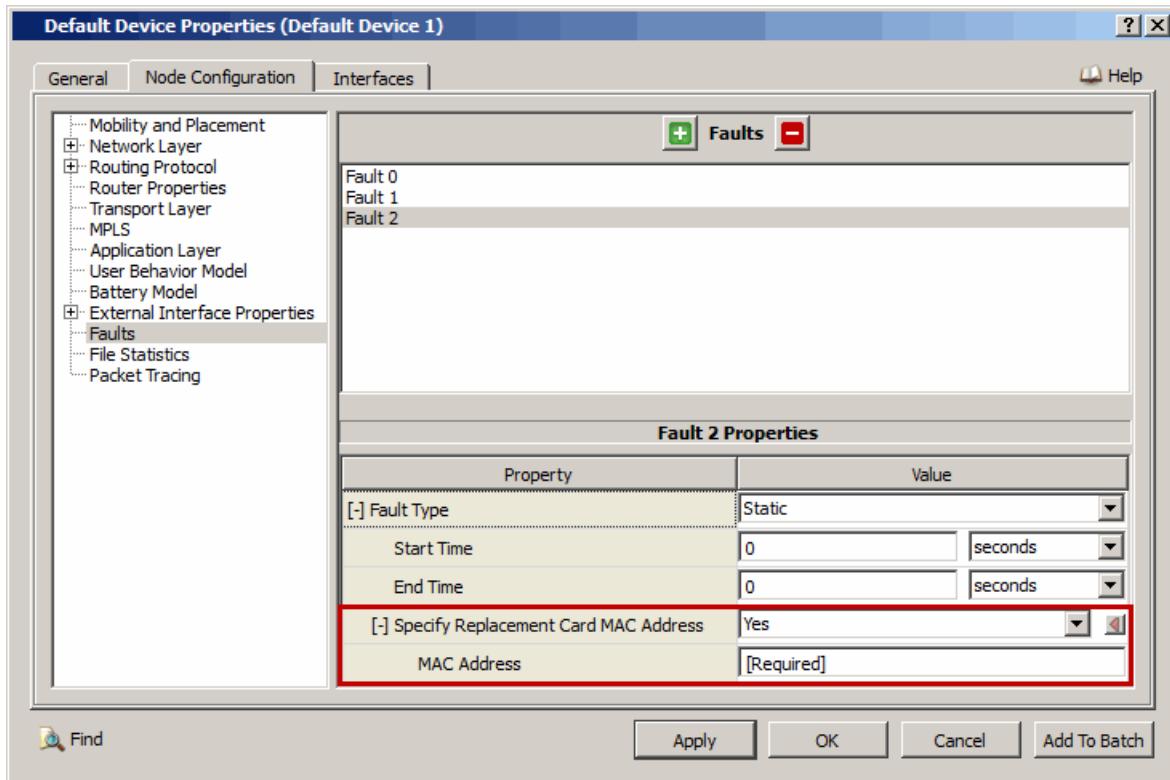


FIGURE 11-4. Specifying Replacement MAC Address

TABLE 11-4. Command Line Equivalent of Specify Replacement Card MAC Address Parameters

| GUI Parameter | Command Line Parameter |
|---------------|-------------------------|
| MAC Address | <interface-mac-address> |

Configuring Dynamic Faults

To configure a dynamic fault, do the following:

1. Set **Fault Type** to *Random*, and set the dependent parameters shown in [Figure 11-5](#).

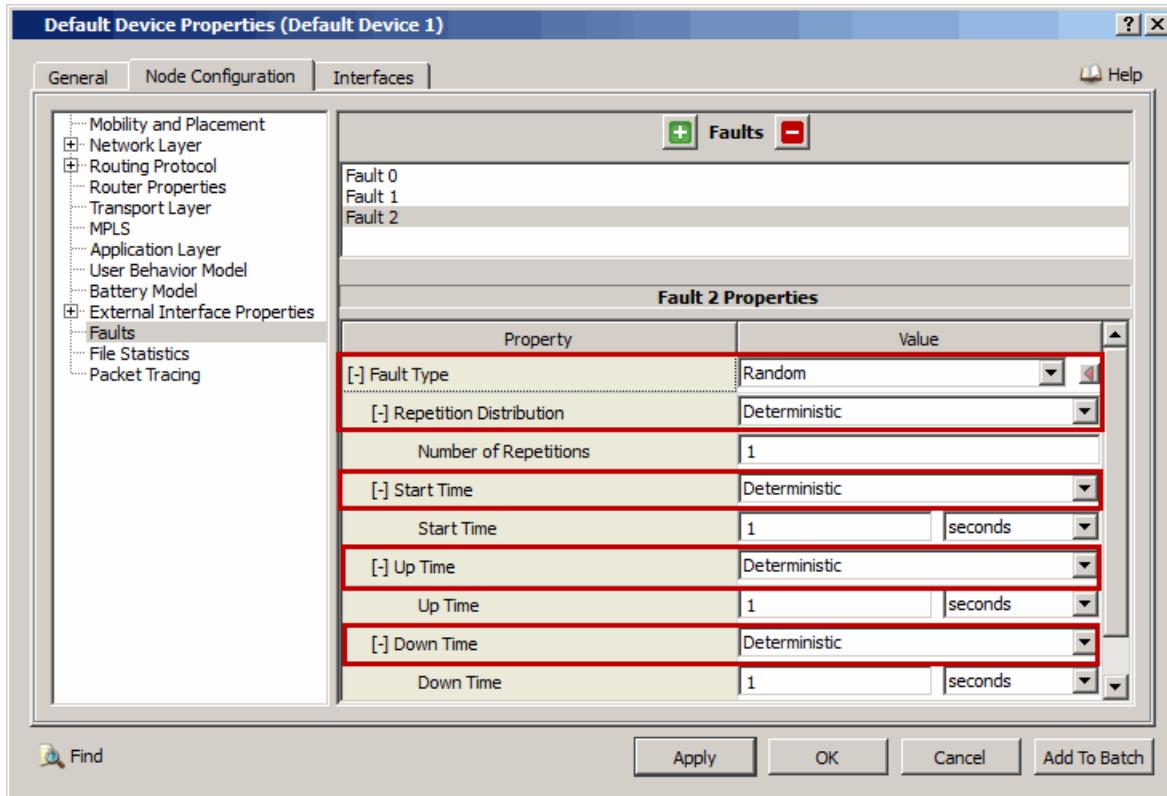


FIGURE 11-5. Setting Dynamic Fault Parameters

TABLE 11-5. Command Line Equivalent of Dynamic Fault Parameters

| GUI Parameter | Command Line Parameter |
|--|------------------------------|
| Repetition Distribution and dependent parameters | REPS <num-repetitions> |
| Start Time and dependent parameters | START <start-time> |
| Up Time and dependent parameters | MTBF <time-between-failures> |
| Down Time and dependent parameters | DOWN <repair-time> |

2. To configure the repetition parameters, set **Repetition Distribution** to *Deterministic*, *Exponential*, or *Uniform*.

- If **Repetition Distribution** is set to *Deterministic*, then set the dependent parameters listed in Table 11-6.

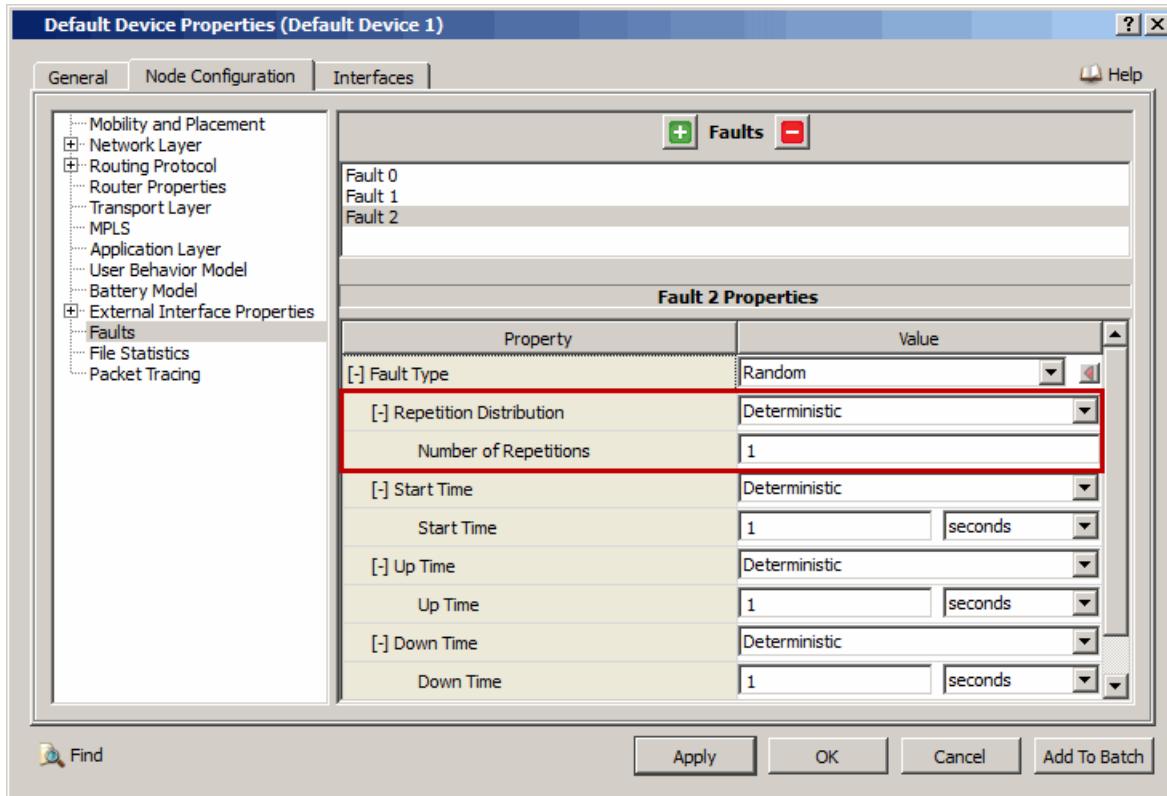


FIGURE 11-6. Setting Parameters for a Deterministic Distribution

TABLE 11-6. Command Line Equivalent of Deterministic Distribution Parameters

| GUI Parameter | Command Line Parameter |
|--|------------------------|
| Repetition Distribution (set to <i>Deterministic</i>) | DET |
| Number of Repetitions | <det-val> |

- If Repetition Distribution is set to *Exponential*, then set the dependent parameters listed in Table 11-7.

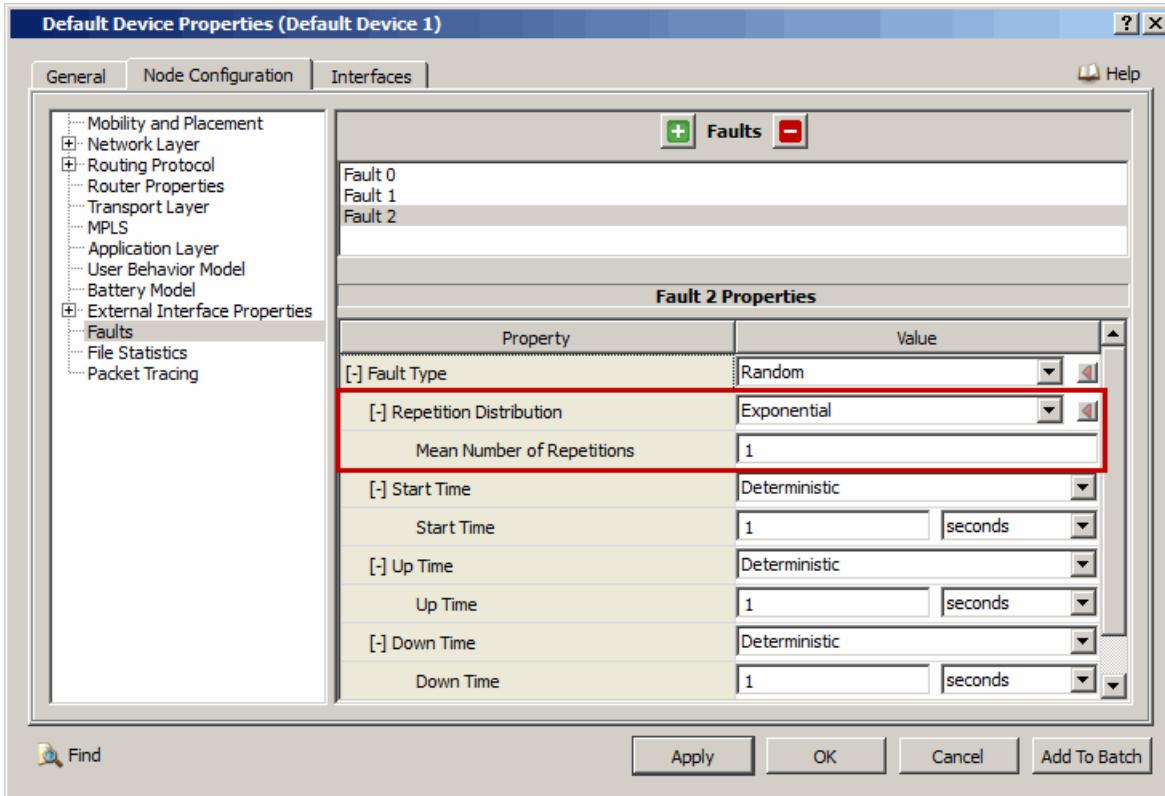


FIGURE 11-7. Setting Parameters for an Exponential Distribution

TABLE 11-7. Command Line Equivalent of Exponential Distribution Parameters

| GUI Parameter | Command Line Parameter |
|--|------------------------|
| Repetition Distribution (set to <i>Exponential</i>) | EXP |
| Mean Number of Repetitions | <exp-val> |

- If Repetition Distribution is set to *Uniform* then set the dependent parameters listed in Table 11-8.

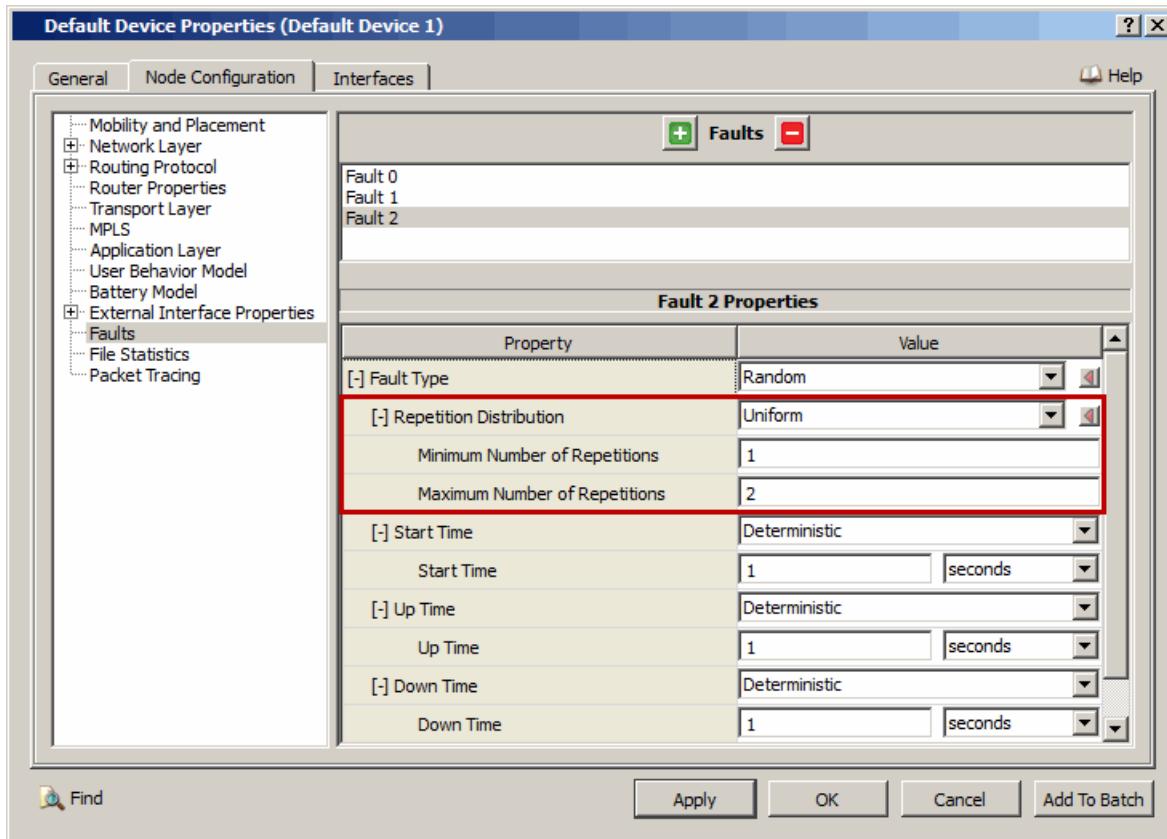


FIGURE 11-8. Setting Parameters for a Uniform Distribution

TABLE 11-8. Command Line Equivalent of Uniform Distribution Parameters

| GUI Parameter | Command Line Parameter |
|--|------------------------|
| Repetition Distribution (set to <i>Uniform</i>) | UNI |
| Minimum Number of Repetitions | <uni-val-1> |
| Maximum Number of Repetitions | <uni-val-2> |

- To configure the start time, set **Start Time** and its dependent parameters in the same way as the repetition parameters.
- To configure the mean time between failures, set **Up Time** and its dependent parameters in the same way as the repetition parameters.
- To configure the mean duration of failures, set **Down Time** and its dependent parameters in the same way as the repetition parameters.

11.1.4 Runtime Visualization

When a scenario is run in Architect's Visualize mode, the following symbols appear dynamically next to nodes indicating the presence of faults:

- A red circle with a diagonal indicates that all interfaces of the node have failed.
- An orange circle with a diagonal indicates that at least one of the interfaces of the node (but not all) has failed.

Refer to the Architect Visualize Mode chapter of *QualNet User's Guide* for details.

11.1.5 Statistics

There are no statistics generated for the Fault models.

11.1.6 Scenarios Included in QualNet

The QualNet distribution includes several sample scenarios for the Fault models. All scenarios are located in the directory `QUALNET_HOME/scenarios/developer/faults`. [Table 11-9](#) lists the sub-directory where each scenario is located.

TABLE 11-9. Fault Models Scenarios Included in QualNet

| Scenario | Description |
|---------------------------|---|
| atm-ip/failure-ip-cloud | Shows whether ATM-IP is working properly when interface failure occurs inside IP-cloud. |
| ripng/sample5 | Shows RIPng behavior with simple split horizon during single interface failure. |
| ripng/sample6 | Shows RIPng behavior during interface failure. |
| ripng/sample8 | Shows RIPng behavior during link failure. |
| trace/drop-interface-down | Shows tracing of the dropped packet when the Interface is down. |

11.2 File-based Node Placement Model

11.2.1 Description

In the File-based Node Placement model, the initial node positions are read from a node position file.

11.2.2 Command Line Configuration

To select the File-based Node Placement model, include the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] NODE-PLACEMENT FILE
```

The scope of this parameter declaration can be Global or Node. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

File-based Node Placement Parameters

[Table 11-10](#) shows the parameters for the File-based Node Placement model. See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 11-10. File-based Node Placement Parameters

| Parameter | Value | Description |
|--|--|--|
| NODE-POSITION-FILE <i>Required</i> Scope: Global, Node <i>Instances:</i> No | Filename | Name of the Node Position file. Note: The same file is also used if the File-based Mobility model is used. The format of the Node Position file is described in Section 11.2.2.1 . |
| MOBILITY-GROUND-NODE <i>Optional</i> Scope: Global, Node | List: • YES • NO <i>Default:</i> NO | Indication whether the node's initial altitude is read from a terrain file. This parameter is used only if a terrain file is also specified in the scenario. YES : The node's z-coordinate (for Cartesian system) or altitude (for Lat-Lon-Alt system) at its initial position is the same as the altitude specified in the terrain file for that point. The value read from the terrain file overrides the z-coordinate or altitude read from the Node Position file. NO : The node's z-coordinate or altitude at its initial position is the one specified in the Node Position file. |

11.2.2.1 Format of the Node Position File

The Node Position file specifies the initial position of each node if the File-based Node Placement model is used. If the File-based Mobility model is used, then this file specifies the node positions at different simulation times.

Each line in the node position file has the following format:

```
<nodeID> <simulation-time> <position>
```

where

| | |
|-------------------|---|
| <nodeID> | Node identifier. |
| <simulation-time> | Simulation time. |
| | For the initial node position, this should be 0. |
| <position> | Node position. The node position is specified as the coordinates in Cartesian or Lat-Lon-Alt system, optionally followed by the orientation (azimuth and elevation). Specifying node orientation is optional and is assumed to be (0.0 0.0) when not specified. Refer to <i>QualNet User's Guide</i> for the format for specifying node positions. |

- Notes:**
1. For each node, the node positions should be sorted (in ascending order) by simulation time.
 2. Each node position specification should be on a single line by itself.
 3. Comments can be entered anywhere in the node position file.

Example

The following lines show a segment of a node position file:

```
1 0 (35.130587432702, -116.72249286971918, 0.0) 0 0
1 10S (35.12977099236641, -116.53095393408505, 0.0) 0.0 0.0
1 20S (35.12977099236641, -116.39738452458609, 0.0)
...
1 60S (35.36132315521628, -116.2700276457615, 0.0)
1 70S (35.465648854961835, -116.26692138042432, 0.0) 0.0 0.0
2 0 (35.16793886702846, -116.72149633406089, 0.0) 0 0
2 10S (35.16897959183674, -116.58344129312579, 0.0) 30.0 0.0
2 20S (35.16938775510204, -116.4518964383234, 0.0) 30.0 45.0
...
```

11.2.3 GUI Configuration

To use the File-based Node Placement model in the GUI, a Node Position file is imported using the Node Placement Wizard. After importing the Node Position file, altitude and orientation parameters for individual nodes can be modified. When the scenario is saved, a file called `<scenario-name>.nodes` is created in the scenario folder. If a file by this name already exists, it is overwritten. This file contains the positions of all nodes on the canvas. When the scenario is run, the simulator uses the File-based placement model and uses `<scenario-name>.nodes` as the Node Position file.

[Section 11.2.3.1](#) describes how to use the Node Placement Wizard to import a node position file. [Section 11.2.3.2](#) describes how to configure node placement parameters for an individual node.

11.2.3.1 Using Node Placement Wizard

To use the File-based Node Placement model in the GUI, do the following:

1. Select **Tools > Node Placement**. This opens the **Node Placement Wizard** shown in Figure 11-9.

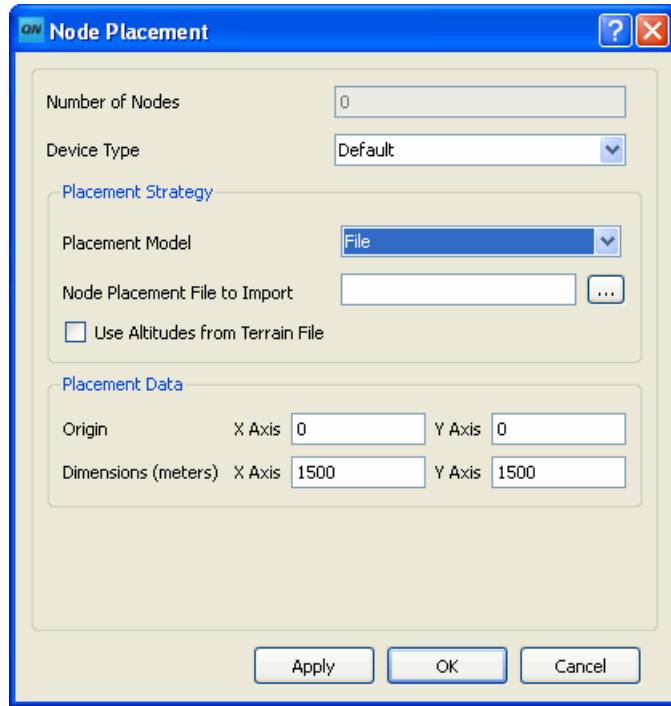


FIGURE 11-9. Node Placement Wizard

2. In the **Number of Nodes** field, enter the desired number of nodes.
3. In the **Device Type** field, select the type of device from the list. All devices that appear in the **Devices** toolbar in the Toolset also appear in the list.
4. Under **Placement Strategy**, set **Placement Model** to *File* and specify the name of the position file in the field **Node Placement File to Import**.
5. Select the check box **Use Altitudes from Terrain File** if terrain elevation data are used in the scenario and the initial altitudes of nodes should be read from the terrain file.
6. Under **Placement Data**, specify the coordinates of the origin and the dimensions of the area in which the nodes are placed (for Cartesian coordinate system) or the latitude and longitude of the south-west and north-east corners of the placement area (for Latitude-Longitude coordinate system).
7. Click **Apply** or **OK**.

TABLE 11-11. Command Line Equivalent of Node Placement Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|--|------------------------|------------------------|
| <i>Node Placement File to Import</i> | Global | NODE-PLACEMENT-FILE |
| <i>Use Altitudes from Terrain File</i> | Global, Node | MOBILITY-GROUND-NODE |

11.2.3.2 Configuring Individual Node Placement Parameters

To configure node placement parameters for a specific node after importing a Node Position file, do the following:

1. Go to **Default Device Properties Editor > Node Configuration > Mobility and Placement**.
2. If terrain elevation data are used in the scenario and the initial altitude of the node should be read from the terrain file, then set **Use Altitudes from Terrain File** to Yes.

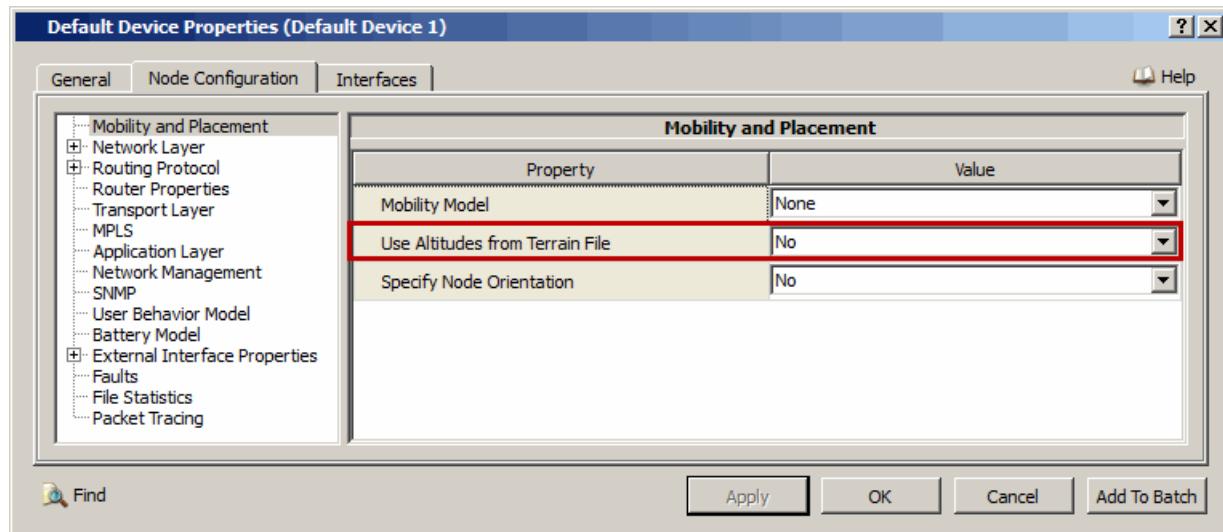


FIGURE 11-10. Setting Node Altitude Parameters

TABLE 11-12. Command Line Equivalent of Node Altitude Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|--|------------------------|------------------------|
| <i>Use Altitudes from Terrain File</i> | Global, Node | MOBILITY-GROUND-NODE |

3. To specify the node orientation, set **Specify Node Orientation** to Yes and set the dependent parameters listed in [Table 11-13](#).

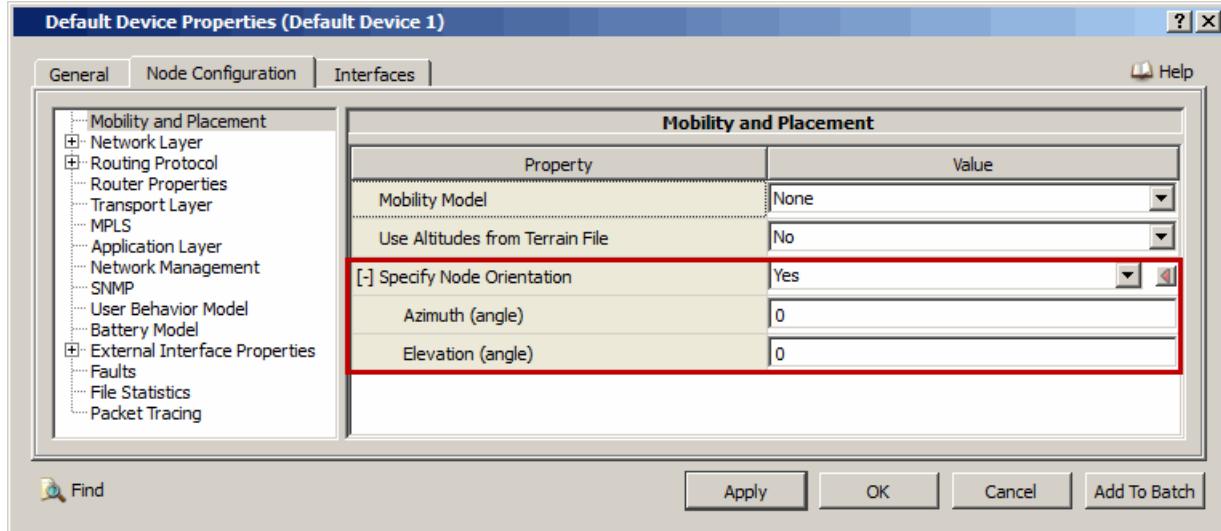


FIGURE 11-11. Setting Node Orientation

TABLE 11-13. GUI Node Orientation Parameters

| GUI Parameter | Description |
|------------------|--|
| <i>Azimuth</i> | Initial (at time 0) azimuth of the node. |
| <i>Elevation</i> | Initial (at time 0) elevation of the node. |

Note: GUI parameters **Azimuth** and **Elevation** do not have direct equivalents in the command line interface. These parameters are written to the Node Placement file that is created (see [Section 11.2.2.1](#) for the format of the Node Position file).

11.3 Grid Node Placement Model

11.3.1 Description

In the Grid Node Placement model, the terrain is divided into a number of squares. One node is placed at each grid point, starting at the origin or the south-west corner. The size of the squares is determined by a user-specified parameter (see [Table 11-14](#)).

11.3.2 Command Line Configuration

To select the Grid Node Placement model, include the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] NODE-PLACEMENT GRID
```

The scope of this parameter declaration can be Global or Node. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

Grid Node Placement Parameters

The parameters for the Grid Node Placement model are described in [Table 11-14](#). See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 11-14. Grid Node Placement Parameters

| Parameter Name | Value | Description |
|--|--|--|
| GRID-UNIT <i>Required</i> Scope: Global | Real <i>Range: > 0</i> <i>Unit: meters or degrees</i> | Length of a side of a square in the grid. If Cartesian coordinate system is used, the unit is meters. If Lat-Lon-Alt system is used, the unit is degrees. |
| MOBILITY-GROUND-NODE <i>Optional</i> Scope: Global, Node | List: • YES • NO <i>Default: NO</i> | Indication whether the node's initial altitude is read from a terrain file. This parameter is used only if a terrain file is also specified in the scenario. YES : The node's z-coordinate (for Cartesian system) or altitude (for Lat-Lon-Alt system) at its initial position is the same as the altitude specified in the terrain file for that point. NO : The node's z-coordinate or altitude at its initial position is 0. |

11.3.3 GUI Configuration

When the Grid Node Placement model is used in the GUI, Architect places nodes within the specified region using a strategy similar to the one described in [Section 11.3.1](#). When the scenario is saved, a file called <scenario-name>.nodes is created in the scenario folder. If a file by this name already exists, it is overwritten. This file contains the positions of all nodes on the canvas. When the scenario is run, the

simulator uses the File-based placement model (see [Section 11.2](#)) and uses <scenario-name>.nodes as the Node Position file.

[Section 11.3.3.1](#) describes how to use the Node Placement Wizard to place nodes on the canvas. [Section 11.3.3.2](#) describes how to configure node placement parameters for an individual node.

11.3.3.1 Using Node Placement Wizard

To use the Grid Node Placement model in the GUI, do the following:

1. Select **Tools > Node Placement**. This opens the **Node Placement Wizard** shown in [Figure 11-12](#).

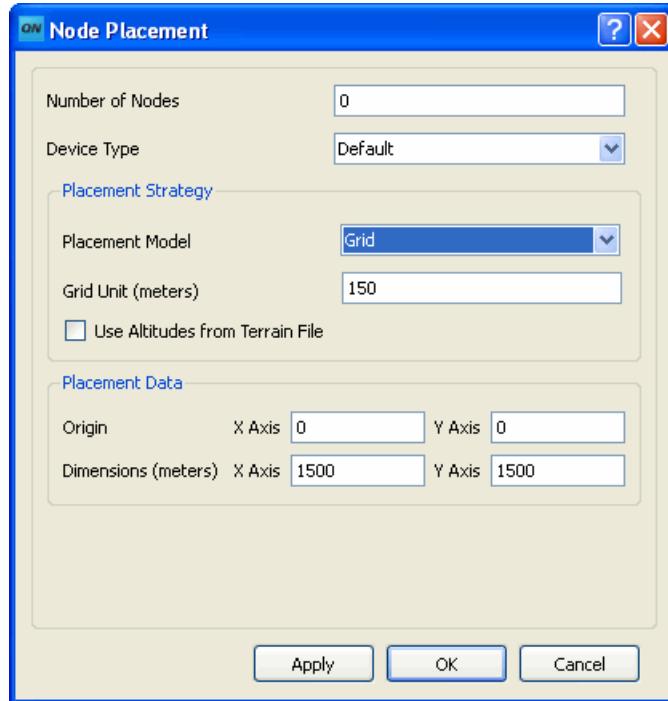


FIGURE 11-12. Node Placement Wizard

2. In the **Number of Nodes** field, enter the desired number of nodes.
3. In the **Device Type** field, select the type of device from the list. All devices that appear in the **Devices** toolbar in the Toolset also appear in the list.
4. Under **Placement Strategy**, set **Placement Model** to **Grid** and specify the grid unit in the **Grid Unit** field.
5. Select the check box **Use Altitudes from Terrain File** if terrain elevation data are used in the scenario and the initial altitudes of nodes should be read from the terrain file.
6. Under **Placement Data**, specify the coordinates of the origin and the dimensions of the area in which the nodes are placed (for Cartesian coordinate system) or the latitude and longitude of the south-west and north-east corners of the placement area (for Latitude-Longitude coordinate system).
7. Click **Apply** or **OK**.

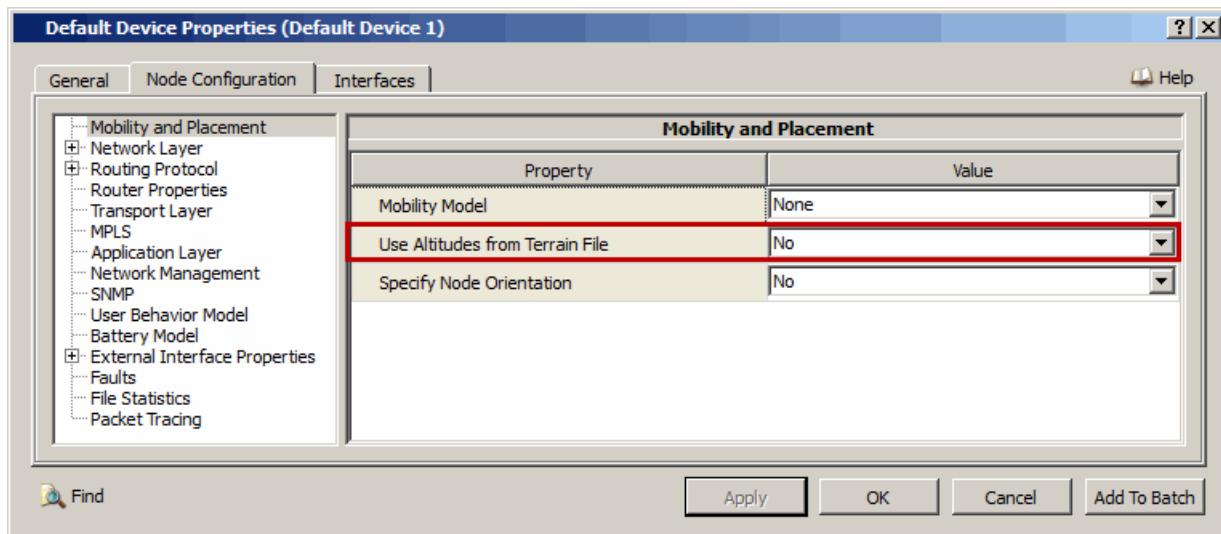
TABLE 11-15. Command Line Equivalent of Node Placement Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|--|------------------------|------------------------|
| <i>Grid Unit</i> | Global | GRID-UNIT |
| <i>Use Altitudes from Terrain File</i> | Global, Node | MOBILITY-GROUND-NODE |

11.3.3.2 Configuring Individual Node Placement Parameters

To configure node placement parameters for a specific node after placing nodes using the Node Placement Wizard, do the following:

1. Go to **Default Device Properties Editor > Node Configuration > Mobility and Placement**.
2. If terrain elevation data are used in the scenario and the initial altitude of the node should be read from the terrain file, then set **Use Altitudes from Terrain File** to Yes.

**FIGURE 11-13. Setting Node Altitude Parameters****TABLE 11-16. Command Line Equivalent of Node Altitude Parameters**

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|--|------------------------|------------------------|
| <i>Use Altitudes from Terrain File</i> | Global, Node | MOBILITY-GROUND-NODE |

3. To specify the node orientation, set **Specify Node Orientation** to Yes and set the dependent parameters listed in [Table 11-17](#).

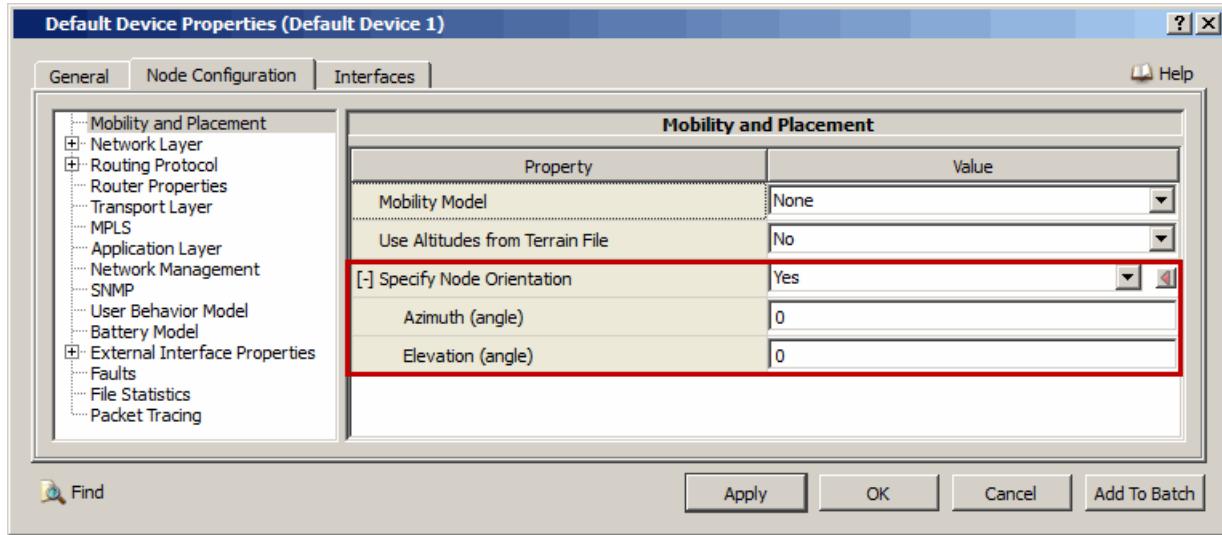


FIGURE 11-14. Setting Node Orientation

TABLE 11-17. GUI Node Orientation Parameters

| GUI Parameter | Description |
|------------------|--|
| <i>Azimuth</i> | Initial (at time 0) azimuth of the node. |
| <i>Elevation</i> | Initial (at time 0) elevation of the node. |

Note: GUI parameters **Azimuth** and **Elevation** do not have direct equivalents in the command line interface. These parameters are written to the Node Placement file that is created (see [Section 11.2.2.1](#) for the format of the Node Position file).

11.4 Random Node Placement Model

11.4.1 Description

In the Random Node Placement model, nodes are placed on the terrain randomly.

11.4.2 Command Line Configuration

To select the Random Node Placement model, include the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] NODE-PLACEMENT RANDOM
```

The scope of this parameter declaration can be Global or Node. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

Random Node Placement Parameters

The parameters for the Random Node Placement model are described in [Table 11-18](#). See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 11-18. Random Node Placement Parameters

| Parameter Name | Value | Description |
|--|--|--|
| MOBILITY-GROUND-NODE <i>Optional</i> Scope: Global, Node | List: • YES • NO <i>Default: NO</i> | Indication whether the node's initial altitude is read from a terrain file. This parameter is used only if a terrain file is also specified in the scenario. YES : The node's z-coordinate (for Cartesian system) or altitude (for Lat-Lon-Alt system) at its initial position is the same as the altitude specified in the terrain file for that point. NO : The node's z-coordinate or altitude at its initial position is 0. |

11.4.3 GUI Configuration

When the Random Node Placement model is used in the GUI, Architect places nodes randomly within the specified region. When the scenario is saved, a file called <scenario-name>.nodes is created in the scenario folder. If a file by this name already exists, it is overwritten. This file contains the positions of all nodes on the canvas. When the scenario is run, the simulator uses the File-based placement model (see [Section 11.2](#)) and uses <scenario-name>.nodes as the Node Position file.

[Section 11.4.3.1](#) describes how to use the Node Placement Wizard to place nodes on the canvas. [Section 11.4.3.2](#) describes how to configure node placement parameters for an individual node.

11.4.3.1 Using Node Placement Wizard

To use the Random Node Placement model in the GUI, do the following:

1. Select **Tools > Node Placement**. This opens the **Node Placement Wizard** shown in Figure 11-15.

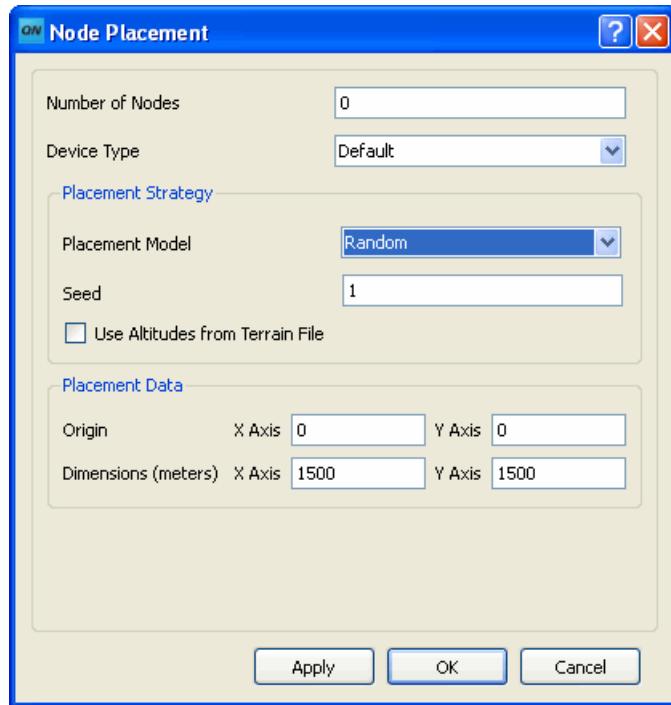


FIGURE 11-15. Node Placement Wizard

2. In the **Number of Nodes** field, enter the desired number of nodes.
3. In the **Device Type** field, select the type of device from the list. All devices that appear in the **Devices** toolbar in the Toolset also appear in the list.
4. Under **Placement Strategy**, set **Placement Model** to *Random* and specify a value for the seed in the **Seed** field. This seed is used to generate random numbers that determine the position of nodes on the canvas.
5. Select the check box **Use Altitudes from Terrain File** if terrain elevation data are used in the scenario and the initial altitudes of nodes should be read from the terrain file.
6. Under **Placement Data**, specify the coordinates of the origin and the dimensions of the area in which the nodes are placed (for Cartesian coordinate system) or the latitude and longitude of the south-west and north-east corners of the placement area (for Latitude-Longitude coordinate system).
7. Click **Apply** or **OK**.

TABLE 11-19. Command Line Equivalent of Node Placement Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|--|------------------------|------------------------|
| Seed | Global | N/A |
| <i>Use Altitudes from Terrain File</i> | Global, Node | MOBILITY-GROUND-NODE |

11.4.3.2 Configuring Individual Node Placement Parameters

To configure node placement parameters for a specific node after placing nodes using the Node Placement Wizard, do the following:

1. Go to **Default Device Properties Editor > Node Configuration > Mobility and Placement**.
2. If terrain elevation data are used in the scenario and the initial altitude of the node should be read from the terrain file, then set **Use Altitudes from Terrain File** to Yes.

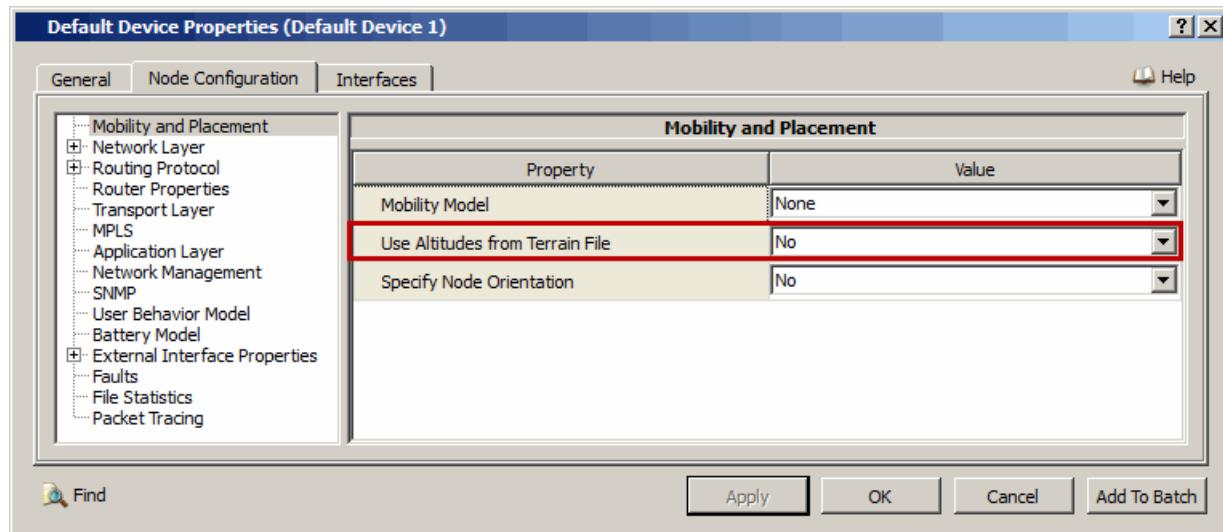


FIGURE 11-16. Setting Node Altitude Parameters

TABLE 11-20. Command Line Equivalent of Node Altitude Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|--|------------------------|------------------------|
| <i>Use Altitudes from Terrain File</i> | Global, Node | MOBILITY-GROUND-NODE |

3. To specify the node orientation, set **Specify Node Orientation** to Yes and set the dependent parameters listed in [Table 11-21](#).

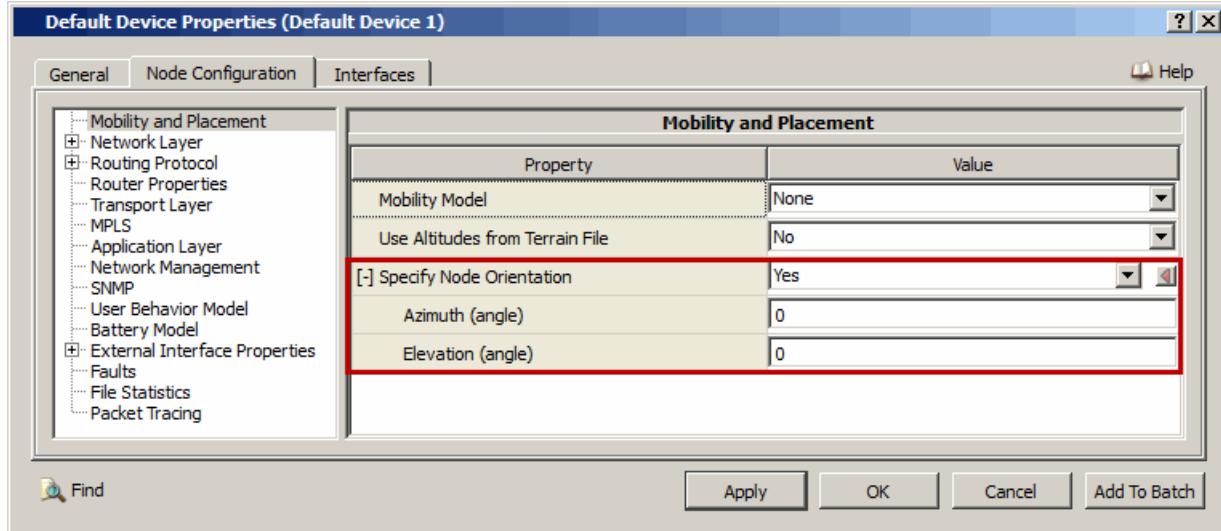


FIGURE 11-17. Setting Node Orientation

TABLE 11-21. GUI Node Orientation Parameters

| GUI Parameter | Description |
|------------------|--|
| <i>Azimuth</i> | Initial (at time 0) azimuth of the node. |
| <i>Elevation</i> | Initial (at time 0) elevation of the node. |

Note: GUI parameters **Azimuth** and **Elevation** do not have direct equivalents in the command line interface. These parameters are written to the Node Placement file that is created (see [Section 11.2.2.1](#) for the format of the Node Position file).

11.5 Uniform Node Placement Model

11.5.1 Description

In the Uniform Node Placement model, the terrain is divided into a number of equal-sized cells, each proportional to the size and shape of the terrain. The number of cells is calculated by rounding the number of nodes to the next higher square of an integer. For example, if there are 23 nodes in a scenario, the terrain is divided into 25 cells. Beginning with the lower left cell, proceeding first along the X-axis, then along the Y-axis, one node is placed at a random position in each cell, until all nodes are placed.

11.5.2 Command Line Configuration

To select the Uniform Node Placement model, include the following parameter in the scenario configuration (.config) file:

```
[<Qualifier>] NODE-PLACEMENT UNIFORM
```

The scope of this parameter declaration can be Global or Node. See [Section 1.2.1.1](#) for a description of <Qualifier> for each scope.

Uniform Node Placement Parameters

The parameters for the Uniform Node Placement model are described in [Table 11-22](#). See [Section 1.2.1.3](#) for a description of the format used for the parameter table.

TABLE 11-22. Uniform Node Placement Parameters

| Parameter Name | Value | Description |
|--|--|--|
| MOBILITY-GROUND-NODE <i>Optional</i> Scope: Global, Node | List: • YES • NO <i>Default:</i> NO | Indication whether the node's initial altitude is read from a terrain file. This parameter is used only if a terrain file is also specified in the scenario. YES : The node's z-coordinate (for Cartesian system) or altitude (for Lat-Lon-Alt system) at its initial position is the same as the altitude specified in the terrain file for that point. NO : The node's z-coordinate or altitude at its initial position is 0. |

11.5.3 GUI Configuration

When the Uniform Node Placement model is used in the GUI, Architect places nodes within the specified region using a strategy similar to the one described in [Section 11.5.1](#). When the scenario is saved, a file called <scenario-name>.nodes is created in the scenario folder. If a file by this name already exists, it is overwritten. This file contains the positions of all nodes on the canvas. When the scenario is run, the simulator uses the File-based placement model (see [Section 11.2](#)) and uses <scenario-name>.nodes as the Node Position file.

Section 11.5.3.1 describes how to use the Node Placement Wizard to place nodes on the canvas. Section 11.5.3.2 describes how to configure node placement parameters for an individual node.

11.5.3.1 Using Node Placement Wizard

To use the Uniform Node Placement model in the GUI, do the following:

1. Select **Tools > Node Placement**. This opens the **Node Placement Wizard** shown in Figure 11-18.

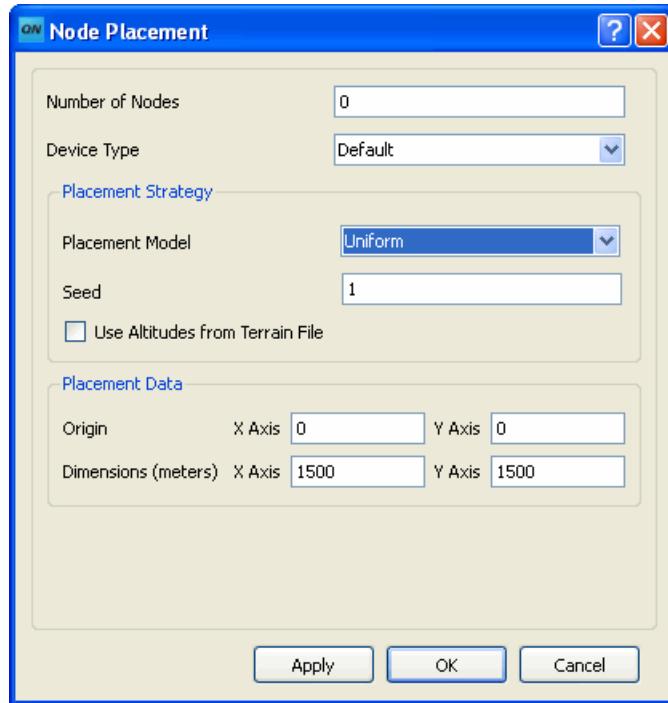


FIGURE 11-18. Node Placement Wizard

2. In the **Number of Nodes** field, enter the desired number of nodes.
3. In the **Device Type** field, select the type of device from the list. All devices that appear in the **Devices** toolbar in the Toolset also appear in the list.
4. Under **Placement Strategy**, set **Placement Model** to *Uniform* and specify a value for the seed in the **Seed** field. This seed is used to generate random numbers that determine the position of nodes on the canvas.
5. Select the check box **Use Altitudes from Terrain File** if terrain elevation data are used in the scenario and the initial altitudes of nodes should be read from the terrain file.
6. Under **Placement Data**, specify the coordinates of the origin and the dimensions of the area in which the nodes are placed (for Cartesian coordinate system) or the latitude and longitude of the south-west and north-east corners of the placement area (for Latitude-Longitude coordinate system).
7. Click **Apply** or **OK**.

TABLE 11-23. Command Line Equivalent of Node Placement Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|--|------------------------|------------------------|
| Seed | Global | N/A |
| <i>Use Altitudes from Terrain File</i> | Global, Node | MOBILITY-GROUND-NODE |

11.5.3.2 Configuring Individual Node Placement Parameters

To configure node placement parameters for a specific node after placing nodes using the Node Placement Wizard, do the following:

1. Go to **Default Device Properties Editor > Node Configuration > Mobility and Placement**.
2. If terrain elevation data are used in the scenario and the initial altitude of the node should be read from the terrain file, then set **Use Altitudes from Terrain File** to Yes.

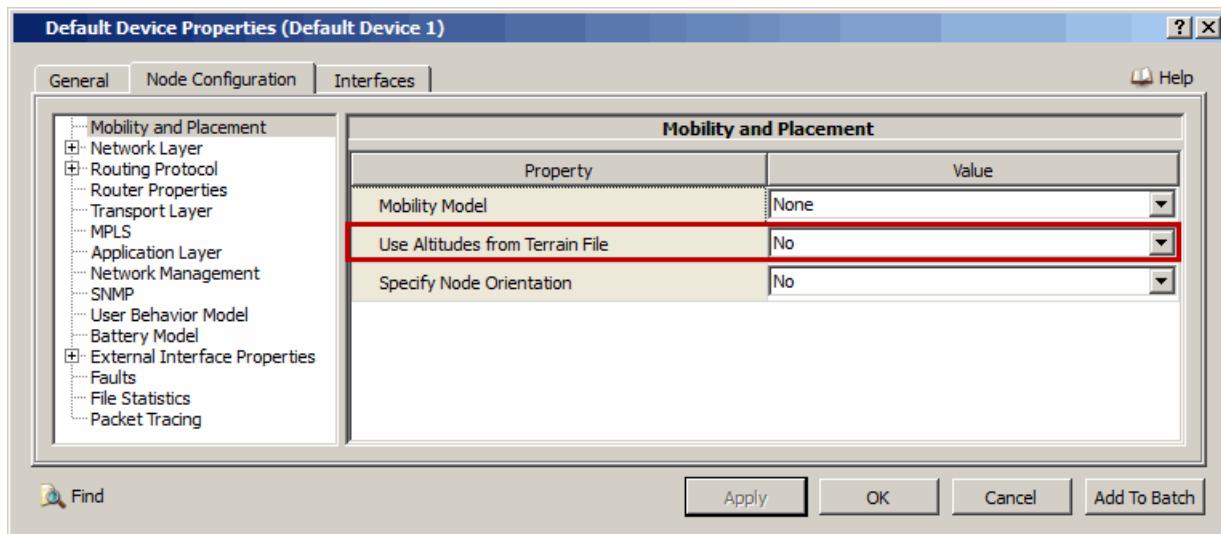


FIGURE 11-19. Setting Node Altitude Parameters

TABLE 11-24. Command Line Equivalent of Node Altitude Parameters

| GUI Parameter | Scope of GUI Parameter | Command Line Parameter |
|--|------------------------|------------------------|
| <i>Use Altitudes from Terrain File</i> | Global, Node | MOBILITY-GROUND-NODE |

3. To specify the node orientation, set **Specify Node Orientation** to Yes and set the dependent parameters listed in [Table 11-25](#).

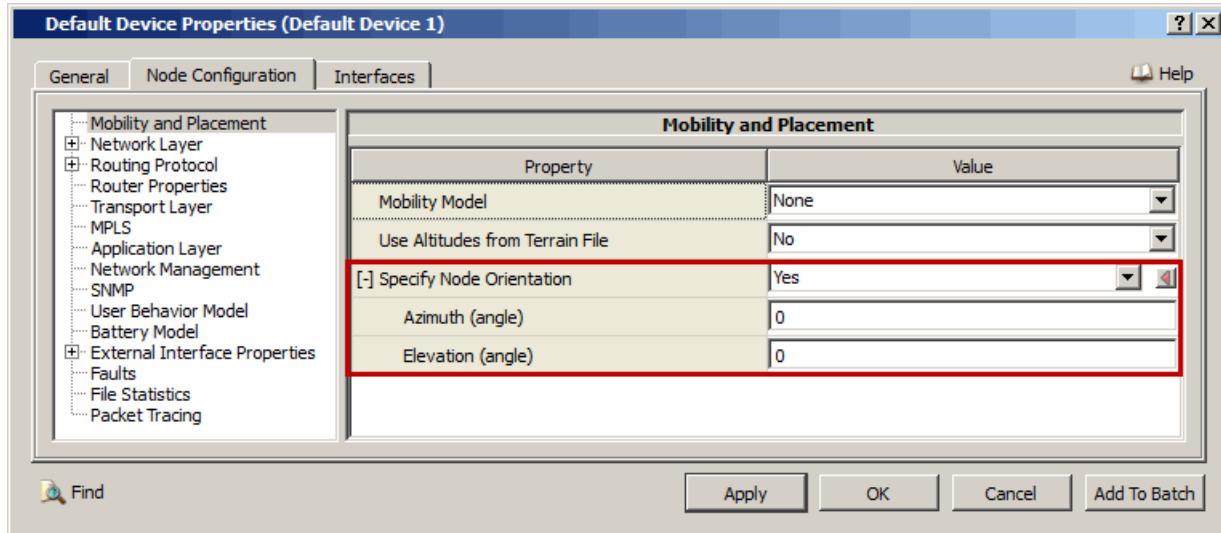


FIGURE 11-20. Setting Node Orientation

TABLE 11-25. GUI Node Orientation Parameters

| GUI Parameter | Description |
|------------------|--|
| <i>Azimuth</i> | Initial (at time 0) azimuth of the node. |
| <i>Elevation</i> | Initial (at time 0) elevation of the node. |

Note: GUI parameters **Azimuth** and **Elevation** do not have direct equivalents in the command line interface. These parameters are written to the Node Placement file that is created (see [Section 11.2.2.1](#) for the format of the Node Position file).