Performance Modeling

and

Knowledge Processing

in

High-Speed Interconnected

Intelligent Educational Networks

by

Marcos Silva Pinto

A Dissertation submitted to the Graduate Faculty in Computer Science in partial fulfillment of the requirements for the degree of Doctor of Philosophy, The City University of New York

2007

UMI Number: 3283160

Copyright 2007 by
Pinto, Marcos Silva

All rights reserved.

UMI Microform 3283160
Copyright 2007 by ProQuest Information and Learning Company.
All rights reserved. This microform edition is protected against unauthorized copying under Title 17, United States Code.

ProQuest Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

©2007

Marcos Silva Pinto

All Rights Reserved

This manuscript has been read and accepted for the Graduate Faculty in Computer Science in satisfaction of the requirement for the degree of Doctor of Philosophy.

April 18, 2007
Date

Professor Syed V. Ahamed
Fellow IEEE
Chair of Examining Committee
The City University of New York

April 18, 2007
Date

Professor Theodore Brown
Executive Officer
The Graduate Center
The City University of New York

April 18, 2007
Date

Professor Michael Kress
Vice-President of Technology Systems
College of Staten Island
The City University of New York

April 18, 2007
Date

Professor Victor B. Lawrence
Associate Dean
Electrical & Computer Engineering
Stevens Institute of Technology

Supervisory Committee
The City University of New York

Abstract

Performance Modeling and Knowledge Processing in High-Speed Interconnected Educational Networks

By

Marcos Silva Pinto

Advisor: Professor Syed V. Ahamed

The objective of all network designers and administrators is to have a network that provides flawlessly the services that users and businesses need. This research will focus on network traffic and customer services in the academic world environment that optimize knowledge processing. The approach is to suggest that once the network traffic behavior is controlled and services can be customized intelligently then the first steps to bring the benefits of the knowledge superhighway to everyone in the world can be taken. First we will analyze traffic characteristics of campus networks in order to develop insight into their operation and performance by developing tractable models and approximations. Then, in the second part of the thesis, we will keep the same optimized campus network structure but now with the addition of intelligent components turning the network into an intelligent educational network which provides knowledge processing. We propose a methodology for understanding network traffic behavior through the combination of quality of service, degradation policies, and a queue scheme. We also suggest a new approach in routing intelligently network packets. The idea is to divide the application layer into layers that deals with the following: the forwarding packet process,

the quality of service integrated into the packet, and finally the type of application the packet belongs to.

The overall purpose of this research is twosome: (i) to optimize high-performance educational networks, implemented as national and global campus networks, by simulating their operations and analyzing their performance based on certain traffic parameters and network characteristics, and (ii) to suggest a general fine-tuned architecture of an intelligent educational systems running on these optimized networks that ultimately acts as a 'negotiator' between a knowledge seeker and knowledge repositories – a matchmaker systems that selects the right set of knowledge data banks (output) after interpreting correctly incoming information request (input).

Acknowledgements

I am very fortunate to have been advised by Professor Syed Vickar Ahamed. He provided me with continuous supervision and encouragement throughout the course of this research. I highly appreciate his suggestions in how to develop a procedural approach to intelligent campus networks.

I am thankful to Professor Theodore Brown, Executive Officer of the Ph/.D. program at the Graduate Center of CUNY for his help during my study of the doctoral program and the directives and valuable suggestions for the completion of this work. I am grateful to Professor Michael Anshel for his assistance and insights on this research. I would also like to thank Professor Michael Kress, Vice President of Information Technology at the College of Staten Island for his advisement and guidance in the core points of the research. I thank Professor Victor Lawrence for his support and encouragement at the final stage of the thesis. Thanks also go to Anil Khullar, Director of Telecommunications of Academic Computing, CUNY, Warren Franklin and Freddy Altamirano, CIS department at City Tech, CUNY, and Kurt Rosenfeld, Electric Engineering department at City College, CUNY; for their positive suggestions and support to improve this work. Appreciation for the all the support should also be given to Professor Candido Cabo, Chair of the Computer Systems Technology at City Tech.

Finally, this study could not have been done without the support of my family: wife Hsinrong for her love, incentive, and patience; our daughter Annie for being curious and forcing me to explain in kids' terms what this work is about, and finally our two sons, Leo and Andrew, for behaving well while 'daddy is working at the computer'.

Performance Modeling and Knowledge Processing in
High-Speed Interconnected Intelligent Educational Networks

Table of Contents

1. Introduction, 1-5
2. Communication Networks, 6
    2.1 Circuit-Switched Networks, 7
        2.1.1 X.25,7-8
    2.2 Packet-Switched Networks, 8-9
3. High-Speed Broadband Networks, 10
    3.1 ISDN, 10-11
    3.2 xDSL, 11
    3.3 SMDS, 12
        3.3.1 SMDS Cell Structure, 12
        3.3.2 SMDS Operation, 13
        3.3.3 SMDS Speeds, 13
    3.4 Frame Relay, 14
        3.4.1 Frame Relay Frames, 14-15
        3.4.2 Frame Relay Operation, 15
        3.4.3 Frame Relay Speeds, 16
    3.5 SONET, 16-17
        3.5.1 SONET Frame, 17-18
        3.5.2 SONET Speeds, 18-19
        3.5.3 SONET Multiplexing, 19-20
        3.5.4 SONET Advantages, 21
    3.6 ATM Networks, 22
        3.6.1 What is an ATM?, 22-23
        3.6.2 How does it Work?, 23-27
        3.6.3 How fast is it?, 28
        3.6.4 ATM Model, 28-29
        3.6.5 ATM Structure, 29
        3.6.6 How ATM Networks Prioritize Traffic, 30
        3.6.7 ATM Cell Switching and Multiplexing, 30-31
        3.6.8 ATM and SONET, 31
4. Campus Networks, 32-35
    4.1 Campus Networks Architectures, 36-37
    4.2 Access Methods in Campus Networks, 37-38
        4.2.1 Point-to-Point Links, 38
    4.3 Information Technologies used in Campus Networks, 39-40
    4.4 Campus Networks and Internet2, 41
        4.4.1 Participating Universities, 42-44
        4.4.2 Key Technologies that Make Up Internet2, 45
        4.4.3 Future Prospects for Internet2, 46
        4.4.4 Abilene Network, 47-50
5. Simulation, 51

5.1 Modes of Simulation, 52
5.2 Time Advancement and the Event List, 52-55
5.3 Event Generation and Random Number Generators, 56
5.4 Transient and Steady States, 57
5.5 Simulation Languages and Tools, 58
5.6 OPNET, 59-60
5.6.1 Modeling using OPNET, 61-63
5.6.2 Model Validation using OPNET, 64
5.6.3 Network Characteristics, 65
5.6.3.1 Application Definition Object, 65
5.6.3.2 Profile Definition Object, 65
6. Queuing Theory, 66-67
6.1 Measures of Queues, 68-70
6.2 Poisson Processes, 70-72
6.3 Queuing Schemas, 73
6.3.1 FIFO, 73
6.3.2 Priority Queue, 73
6.3.3 Weighted Fair Queuing, 73-74
6.3.4 Class-Based Queuing, 74
7. Quality of Service, 75-77
7.1 QoS in IP Network, 78-81
7.1.1 IntServ Model, 82
7.1.2 DiffServ Model, 82-85
7.1.3 MPLS Model, 85
8. Network Performance, 86
8.1 Network Metric Fundamentals, 87
8.1.1 Circuits (Links), 88
8.1.2 Utilization, 88
8.1.3 Throughput, 89
8.1.4 Queue Delay, 90
8.1.5 Response Time, 90
8.1.6 End-To-End Delay, 90-93
9. The Interconnect Campus Network, 94
9.1 Geographic Scope, 95-96
9.2 Modeling and Simulation of an Interconnected Campus Network, 97
9.3 Methodology used for Simulation the interconnected campus network, 97-99
10. National and International Campus Networks (NCN & ICN), 100-102
10.1 Network Traffic, 102-105
10.2 Subnets, 105-107
10.2.1 Switch, 107
10.2.2 Router, 107-108
10.2.3 Links, 108-109
10.2.4 Server, 109
10.2.5 LAN, 109-110
10.3 Configurations Characteristics, 110-112
10.3.1 User's Characteristics, 112-113

10.3.2 Applications Characteristics, 114
10.4 Simulation Scenarios, 115
10.5 Network Links, 116
10.6 Background Traffic, 116-117
10.7 Sensitivity Test, 117
10.8 Run-Time, 118
11. Simulation Results, 118-120
11.1 NCN – Scenario 11: Link, 120
11.1.1 End-To-End Delay, 121-122
11.1.2 Response Times, 123-124
11.1.3 Queue Delay, 125-126
11.1.4 Throughput, 127-128
11.1.5 Utilization, 129-130
11.2 NCN – Scenario 1F: Link Failure, 131
11.2.1 End-To-End Delay, 132
11.2.2 Response Times, 132-133
11.2.3 Queue Delay, 133-134
11.2.4 Throughput, 134-135
11.2.5 Utilization, 135-136
11.2.6 Analysis of the Graphs, 136-137
11.3 NCN – Scenario 13: Link + Backload + QoS, 138
11.3.1 End-To-End Delay, 139-140
11.3.2 Response Times, 141-147
11.3.3 Queue Delay, 148-150
11.3.4 Throughput, 151-152
11.3.5 Utilization, 153-154
11.4 ICN – Scenario 21: Link, 155-157
11.4.1 End-To-End Delay, 158-159
11.4.2 Response Times, 160-162
11.4.3 Queue Delay, 163-165
11.4.4 Throughput, 166-168
11.4.5 Utilization, 169-170
11.5 ICN – Scenario 22: OC3 + Backload + QoS, 172-173
11.5.1 End-To-End Delay, 174-175
11.5.2 Response Times, 176-177
11.5.3 Queue Delay, 178-180
11.5.4 Throughput, 181-184
11.5.5 Utilization, 185-187
11.6 ICN – Scenario 23: Link + Backload + QoS, 188
11.6.1 End-To-End Delay, 189
11.6.2 Response Times, 190-192
11.6.3 Queue Delay, 193-196
11.6.4 Throughput, 197-199
11.6.5 Utilization, 200-202
11.7 Real Network Traffic, 203
11.7.1 Actual Throughput – Abilene Peers, 204-207

11.7.2 Actual Round-Trip Time – Abilene Peers, 208
12. Intelligence in Networks, 209
13. Intelligent Network (IN), 210-212
13.1 INCM Model, 212
13.2 IN Components, 215
13.2.1 Service Switching Point (SSP), 216
13.2.2 Service Control Point (SCP), 216
13.2.3 Service Management System (SMS), 217
13.2.4 Intelligent Peripheral (IP), 217
13.2.5 Signaling Network (SS7), 217-220
13.2.6 Service Transfer Point (STP), 220
13.3 Intelligent Network 1 (IN/1), 221
13.4 Intelligent Network 2 (IN/2), 222
13.5 Advanced Intelligent Network (AIN), 223-224
13.6 Convergence IN and IP Networks, 225-227
14. Intelligent Educational Systems (IES), 228-230
14.1 Characteristics of IES, 231-232
14.2 Functions of the Intelligent Processing Layer, 233
14.3 Structure of Knowledge Base System (KBS), 233-234
14.4 Knowledge Domain According to Dewey Decimal Code (DDC), 235-236
14.5 IES Model Design, 237
14.6 Proposed IES Model, 238-242
14.7 Simulation of IES, 243
14.7.1 IES Topology, 244-246
14.7.2 Representing the Network Traffic, 247-254
14.7.3 Network Simulation Operations, 255-257
14.7.4 Running the Simulation, 258
14.7.5 Scenario of Intelligent National Campus Network, 258-260
14.7.6 DB Query Statistics, 261-267
14.7.7 HTTP Statistics, 268-271
14.7.8 Email Statistics, 272-275
14.7.9 TCP Statistics, 276
14.7.10 IP Statistics, 277
14.7.11 Ethernet Statistics, 278
14.7.12 Link Statistics, 279-293
14.7.13 Analysis of the Simulation Results, 294-297
15. Conclusion and Future Research, 298-301
15.1 Future Research, 302-304
16. Glossary, 305-311
17. Bibliography, 312-315

Performance Modeling and Knowledge Processing in
High-Speed Interconnected Intelligent Educational Networks

List of Tables

Table 1 : SONET Speed Hierarchy, 19

Table 2 : Advantages of using SONET, 21

Table 3 : High-Speed Wide Area Connection Technologies, 28

Table 4 : Categories of Technology used in Education, 39

Table 5 : 2004 Internet2 Membership Fees, 42

Table 6 : Abilene's International Peer Networks, 50

Table 7 : Quality of Service Requirements, 77

Table 8 : Heavy-user Applications Parameters, 113

Table 9: ToS values, 114

Table 10: Simulation Scenarios, 115

Table 11: Backload Traffic Levels, 116

Table 12: NCN: Traffic Load, 120

Table 13: Mean Response Times, 124

Table 14: NCN Link Utilization, 130

Table 15: ICN: Network Applications Traffic Load, 157

Table 16: Applications Response Times, 162

Table 17: ICN Subnet Throughput, 168

Table 18: ICN Link Utilization, 171

Table 19: NCN & ICN Applications Response Times, 177

Table 20: ICN & NCN Queue Delay, 180

Table 21: ICN & NCN Throughput, 184

Table 22: ICN Subnets Queue Delay, 196

Table 23: ICN Subnets Throughput, 199

Table 24: ICN Subnets Link Utilization, 202

Table 25: Top-Level Dewey Decimal System, 235

Table 26: Database Query Attributes, 248

Table 27: Online Searches by DDC, 249

Table 28: ToS Values, 253

Table 29: Global Statistics for DB-Query, 261

Table 30: Node Statistics – DB Query, 261

Table 31: Values of Global Statistics Values – DB Query, 262

Table 32: NY Node: DB-Query Statistics, 265

Table 33: LA Node: DB-Query Statistics, 267

Table 34: Global Statistics for HTTP, 268

Table 35: Node Statistics for HTTP, 268

Table 36: Values of Global & Node Statistics: HTTP, 271

Table 37: Global Statistics for E-Mail, 272

Table 38: Node Statistics for E-Mail, 272

Table 39: Values of Global & Node Statistics: E-Mail, 275

Table 40: Global Statistics for TCP, 276

Table 41: Values of Global Statistics: TCP Delay (sec), 276

Table 42: Global Statistics for IP, 277

Table 43: Values of Global Statistic: IP Traffic Dropped (packets/sec), 277

Table 44: Global Statistics for Ethernet, 278

Table 45: Values Global Statistics: Ethernet Delay (sec), 278

Table 46: Link (Port-to-Port) Statistics, 279

Table 47: Link Statistics: New York – Miami, 281

Table 48: Link Statistics: Miami – Houston, 283

Table 49: Link Statistics: Houston – Chicago, 285

Table 50: Link Statistics: Houston – Los Angeles, 287

Table 51: Link Statistics Los Angeles – Seattle, 289

Table 52: Link Statistics Seattle – Chicago, 291

Table 53: Link Statistics Chicago – New York, 293

List of Figures

Figure 1 : Generation & Propagation of Community Intelligence, 1

Figure 2 : X.25 Network, 8

Figure 3 : SMDS Cell Structure, 12

Figure 4 : Frame Relay Frame, 15

Figure 5 : SDH Global Standards, 16

Figure 6 : SONET Frame, 18

Figure 7 : SONET Multiplexer Operation, 20

Figure 8 : ATM support of various traffic types, 23

Figure 9 : ATM Cell Format, 24

Figure 10: Establishing a connection in an ATM network, 24

Figure 11: ATM Virtual Paths and Virtual Channels, 26

Figure 12: ATM Switch Operations, 26

Figure 13: Virtual Circuit and Virtual Path Switching, 27

Figure 14: ATM Model, 28

Figure 15: ATM Network, 29

Figure 16: Typical Campus Network, 34

Figure 17: Layer Design of a Network, 35

Figure 18: Causal Relationship between Elements of an Educational Systems, 40

Figure 19: Two Campus Networks Interconnected via Backbones, 43

Figure 20: Inside the Pacific/Northwest GigaPoP, 44

Figure 21: Abilene domestic peers, 47

Figure 22: Abilene International Network Peers, 49

Figure 23 : System Predictability vs System Complexity, 51

Figure 24 : Sequence of Operations in an Event Driven Simulator, 55

Figure 25 : Simulation Development Process with OPNET, 61

Figure 26 : Queuing System, 66

Figure 27 : Poisson distribution, 71

Figure 28 : Abilene Weekly Traffic by Application Type, 76

Figure 29 : IPv4 Frame Structure, 79

Figure 30 : National Campus Network, 95

Figure 31 : International Campus Network, 96

Figure 32: Node Layout of National Campus Network, 100

Figure 33: Node Layout of International Campus Network, 101

Figure 34 : NY subnet, 106

Figure 35 : Daily traffic of Brooklyn College, CUNY, 111

Figure 36 : Daily traffic of CUNY, 112

Figure 37 : NCN: Traffic load comparison, 119

National Campus Network – Scenario 11 - Links

Figure 38 : NCN: TCP delay, 121

Figure 39 : NCN: Ethernet delay, 121

Figure 40: NCN: E-Mail response time, 123

Figure 41: NCN: FTP response time, 123

Figure 42: NCN: HTTP response time, 123

Figure 43: NCN: DB Query response time, 123

Figure 44: NCN: Queue delay Hou-Chi, 125

Figure 45: NCN: Queue delay Hou-Chi no T-3, 125

Figure 46: NCN: Queue delay LA-Seattle, 125

Figure 47: NCN: Queue delay NY-Miami, 125

Figure 48: NCN: Throughput Hou-Chi, 127

Figure 49: NCN: Throughput LA-Seattle, 127

Figure 50: NCN: Throughput NY-Miami, 127

Figure 51: NCN: IP Traffic Dropped, 127

Figure 52: NCN: Utilization Hou-Chi, 129

Figure 53: NCN: Utilization LA-Seattle, 129

Figure 54: NCN: Utilization NY-Miami, 129

National Campus Network – Scenario 1F: T-3 Link, Device Failure

Figure 55 NY node – Failed Distribution Switch 1, 131

Figure 56: NCN: TCP delay, 132

Figure 57: NCN: Ethernet delay, 132

Figure 58: NCN: E-mail response time, 132

Figure 59: NCN: FTP response time, 132

Figure 60: NCN: HTTP response time, 133

Figure 61: NCN: DB Query response time, 133

Figure 62 : NCN: Queue delay Hou-Chi, 133

Figure 63 : NCN: Queue delay LA-Seattle, 133

Figure 64 : NCN: Queue delay NY-Miami, 134

Figure 65 : NCN: Throughput Hou-Chi, 134

Figure 66 : NCN: Throughput LA-Seattle, 134

Figure 67 : NCN: Throughput NY-Miami, 135

Figure 68 : NCN: Utilization Hou-Chi, 135

Figure 69 : NCN: Utilization LA-Seattle, 135

Figure 70 : NCN: Utilization NY-Miami, 136

National Campus Network – Scenario 13 – Link + Backload + QoS

Figure 71 : NCN: TCP delay T3/OC3, 139

Figure 72 : NCN: TCP delay OC12/OC48, 139

Figure 73 : NCN: Ethernet delay T3/OC3, 139

Figure 74 : Ethernet delay OC12/OC48, 139

Figure 75 : NCN: E-mail response time T3, 141

Figure 76 : NCN: E-mail response time OC3, 141

Figure 77 : NCN: E-mail response time OC12, 141

Figure 78 : NCN: E-mail response time OC48, 141

Figure 79 : NCN: FTP response time T3, 142

Figure 80 : NCN: FTP response time OC3, 142

Figure 81 : NCN: FTP response time OC12, 142

Figure 82 : NCN: FTP response time OC48, 142

Figure 83 : NCN: HTTP response time T3, 143

Figure 84 : NCN: HTTP response time OC3, 143

Figure 85 : NCN: HTTP response time OC12, 143

Figure 86 : NCN: HTTP response time OC48, 143

Figure 87 : NCN: DB Query T3, 144

Figure 88 : NCN: DB Query OC3, 144

Figure 89 : NCN: DB Query OC12, 144

Figure 90 : NCN: DB Query OC48, 144

Figure 91 : NCN: Queue delay Hou-Chi T3/OC3, 148

Figure 92 : NCN: Queue delay Hou-Chi OC12/OC48, 148

Figure 93 : NCN: Queue delay LA-Seattle T3/OC3, 148

Figure 94 :NCN: Queue delay LA-Seattle OC12/OC48, 148

Figure 95 NCN: Queue delay NY-Miami T3/OC3, 149

Figure 96: NCN: Queue delay NY-Miami OC12/OC48, 149

Figure 97: NCN: Throughput Hou-Chi T3/OC3, 151

Figure 98: NCN: Throughput Hou-Chi OC12/OC48, 151

Figure 99: NCN: Throughput LA-Seattle T3/OC3, 151

Figure 100: NCN: Throughput LA-Seattle OC12/OC48, 151

Figure 101: NCN: Throughput NY-Miami T3/OC3, 152

Figure 102: NCN: Throughput NY-Miami OC12/OC48, 152

Figure 103: NCN: Utilization Hou-Chi T3/OC3, 153

Figure 104: NCN: Utilization Hou-Chi OC12/OC48, 153

Figure 105: NCN: Utilization LA-Seattle T3/OC3, 153

Figure 106: NCN: Utilization LA-Seattle OC12/OC48, 153

Figure 107: NCN: Utilization NY-Miami T3/OC3, 154

Figure 108: NCN: Utilization NY-Miami OC12/OC48, 154

International Campus Network – Scenario 21 - Links

Figure 109: ICN: Traffic load comparison, 156

Figure 110: ICN: TCP delay, 158

Figure 111: ICN: TCP delay – no T3, 158

Figure 112: ICN: Ethernet delay, 158

Figure 113: ICN: IP Packets Dropped, 158

Figure 114: ICN: E-mail, 160

Figure 115: ICN: E-mail – no T3, 160

Figure 116: ICN: FTP, 160

Figure 117: ICN: FTP – no T3, 160

Figure 118: ICN: HTTP, 161

Figure 119: ICN: HTTP – no T3, 161

Figure 120: ICN: DB Query, 161

Figure 121: ICN: DB Query – no T3, 161

Figure 122: ICN: Queue delay Hou-Chi, 163

Figure 123: ICN: Queue delay LA-Seattle, 163

Figure 124: ICN: Queue delay NY-Miami – T3, 163

Figure 125: ICN: Queue delay NY-Miami – no T3, 163

Figure 126: ICN: Queue delay NY-London, 164

Figure 127: ICN: Queue delay LA-Seattle, 164

Figure 128: ICN: Queue delay NY-London no T3/OC3, 164

Figure 129: ICN: Queue delay Miami-Sao Paulo, 164

Figure 130: ICN: Queue delay LA-Tokyo, 165

Figure 131: ICN: Throughput Houston-Chicago, 166

Figure 132: ICN: Throughput LA-Seattle, 166

Figure 133: ICN: Throughput NY-Miami – T3, 166

Figure 134: ICN: Throughput NY-Miami – no T3, 166

Figure 135: ICN: Throughput NY-London, 167

Figure 136: ICN: Throughput Miami-Sao Paulo, 167

Figure 137: ICN: Throughput Miami-Sao Paulo no T3, 167

Figure 138: ICN: Throughput LA-Tokyo, 167

Figure 139: ICN: Utilization Houston-Chicago, 169

Figure 140: ICN: Utilization LA-Seattle, 169

Figure 141: ICN: Utilization NY-Miami – T3, 169

Figure 142: ICN: Utilization NY-Miami no T3, 169

Figure 143: ICN: Utilization NY-London, 170

Figure 144: ICN: Utilization Miami-Sao Paulo, 170,

Figure 145: ICN: Utilization Miami-Sao Paulo no T3, 170

Figure 146: ICN: Utilization LA-Tokyo, 180

International Campus Network – Scenario 23 – OC-3 Link + Backload + QoS

Figure 147: ICN: TCP delay (as is), 174

Figure 148: ICN: TCP delay, 174

Figure 149: ICN: Ethernet delay (as is), 174

Figure 150: ICN: Ethernet delay, 174

Figure 151: ICN: E-mail, 176

Figure 152: ICN: FTP, 176

Figure 153: ICN: HTTP, 176

Figure 154: ICN: DB Query, 176

Figure 155: ICN: Queue delay Houston-Chicago, 178

Figure 156: ICN: Queue delay LA-Seattle, 178

Figure 157: ICN: Queue delay NY-Miami, 178

Figure 158: ICN: Queue delay NY-London, 178

Figure 159: ICN: Queue delay Miami-Sao Paulo, 179

Figure 160: ICN: Queue delay LA-Tokyo, 179

Figure 161: ICN: Throughput Houston-Chicago, 181

Figure 162: ICN: Throughput LA-Seattle, 181

Figure 163: ICN: Throughput NY-Miami, 181

Figure 164: ICN: Throughput NY-London, 181

Figure 165: ICN: Throughput Miami-Sao Paulo (as is), 182

Figure 166: ICN: Throughput Miami-Sao Paulo, 182

Figure 167: ICN: Throughput Miami-Sao Paulo OC3 only, 182

Figure 168: ICN: Throughput LA-Tokyo (as is), 182

Figure 169: ICN: Throughput LA-Tokyo, 183

Figure 170: ICN: IP traffic dropped, 183

Figure 171: ICN: Utilization Houston-Chicago, 185

Figure 172: ICN: Utilization LA-Seattle, 185

Figure 173: ICN: Utilization NY-Miami, 185

Figure 174: ICN: Utilization NY-London, 185

Figure 175: ICN: Utilization Miami-Sao Paulo, 186

Figure 176: ICN: Utilization Miami-Sao Paulo OC3 only, 186

Figure 177: ICN: Utilization LA-Tokyo, 186

International Campus Network – Scenario 23 – Link (no T3) + Backload + QoS

Figure 178: ICN: TCP delay, 189

Figure 179: ICN: Ethernet delay, 189

Figure 180: ICN: E-mail, 190

Figure 181: ICN: FTP, 190

Figure 182: ICN: HTTP, 190

Figure 183: ICN: DB Query, 190

Figure 184: ICN: Queue delay Houston-Chicago, 193

Figure 185: ICN: Queue delay Houston-Chicago no OC3, 193

Figure 186: ICN: Queue delay LA-Seattle, 193

Figure 187: ICN: Queue delay NY-Miami, 193

Figure 188: ICN: Queue delay NY-Miami no OC3, 194

Figure 189: ICN: Queue delay NY-London, 194

Figure 190: ICN: Queue delay NY-London no OC3, 194

Figure 191: ICN: Queue delay Miami-Sao Paulo, 194

Figure 192: ICN: Queue delay LA-Tokyo, 195

Figure 193: ICN: Throughput Houston-Chicago, 197

Figure 194: ICN: Throughput LA-Seattle, 197

Figure 195: ICN: Throughput NY-Miami, 197

Figure 196: ICN: Throughput NY-London, 197

Figure 197: ICN: Throughput Miami-Sao Paulo, 198

Figure 198: ICN: Throughput LA-Tokyo, 198

Figure 199: ICN: Utilization Houston-Chicago, 200

Figure 200: ICN: Utilization LA-Seattle, 200

Figure 201: ICN: Utilization NY-Miami, 200

Figure 202: ICN: Utilization NY-London, 200

Figure 203: ICN: Utilization Miami-Sao Paulo, 201

Figure 204: ICN: Utilization LA-Tokyo, 201

Actual Traffic of some Abilene Peer Networks

Figure 205: Actual Traffic NY-London, 204

Figure 206: Actual Traffic Miami-Sao Paulo, 204

Figure 207: Total Simulated Traffic Miami-Sao Paulo, 205

Figure 208: Actual Traffic LA-Tokyo, 206

Figure 209: Actual Traffic NY-Chicago, 207

Figure 210: Total Simulated Traffic NY-Chicago, 207

Figure 211: Actual Round-Trip Time Miami-Sao Paulo, 208

Intelligent Campus Network

Figure 212: 4-Plane Structure of IN Model, 214

Figure 213: IN Components, 215

Figure 214: In-Bound Signaling, 218

Figure 215: Out-Bound Signaling, 219

Figure 216: 4-Layer Structure of SS& vs 7-Layer OSI, 219

Figure 217: Intelligent Network 1 (IN/1), 221

Figure 218: Components of Advanced Intelligent Network (AIN), 223

Figure 219: Voice/Data Integration to the Desktop, 226

Figure 220: Conceptual Diagram of IENA, 228

Figure 221: Layer Structure of IES, 232

Figure 222: Components of a Knowledge Base System, 234

Figure 223: Reynold's Design Triangle, 237

Figure 224: DB Query Processing by an IES, 238

Figure 225: Conceptual Model of an IES, 239

Figure 226: Layout of the Intelligent National Campus Network, 244

Figure 227: Subnet Nodes with Interfaces acting as SCP, 245

Figure 228: Database Application Settings, 248

Figure 229: Profile Configuration, 250

Figure 230: Conceptual View of Packet Forwarding with QoS & Content-Inspection, 256

Figure 231: Algorithm for Processing Database Queries, 257

Figure 232: INCN Traffic Flow through New York Node, 260

Figure 233: DB Query Response Time (sec), 262

Figure 234: DB Query Traffic Sent (bytes/sec), 262

Figure 235: DB Query Traffic Received (bytes/sec), 262

Figure 236: DB- Tech&Appli Load (requests/sec), 263

Figure 237: DB-NatSci&Math Load(requests/sec), 263

Figure 238: DB-Hist&Geo Load (requests/sec), 263

Figure 239: DB-Tech&Appli Traffic Sent (bytes/sec), 264

Figure 240: DB-NatSci&Math Traffic Sent (bytes/sec), 264

Figure 241: DB-Hist&Geo Traffic Received (bytes/sec), 264

Figure 242: DB-Hist&Geo Traffic Sent (bytes/sec), 264

Figure 243: DB-Tech&Appli Traffic Recvd (bytes/sec), 265

Figure 244: DB-NatSci&Math Traffic Recvd (bytes/sec), 265

Figure 245: DB-Religion Load (requests/sec), 266

Figure 246: DB-Philo&Psycho Load (requests/sec), 266

Figure 247: DB-Religion Traffic Sent (bytes/sec), 266

Figure 248: DB-Phi&Psy Traffic Sent (bytes/sec), 266

Figure 249: DB-Religion Traffic Recvd (bytes/sec), 267

Figure 250: DB-Phi&Psy Traffic Recvd (bytes/sec), 267

Figure 251: INCN HTTP Object Response Time (sec), 269

Figure 252: INCN HTTP Page Response Time (sec), 269

Figure 253: INCN HTTP Traffic Sent (bytes/sec), 269

Figure 254: INCN HTTP Traffic Received (bytes/sec), 269

Figure 255: INCN HTTP Traffic Sent NY (bytes/sec), 270

Figure 256: INCN HTTP Traffic Sent Miami (bytes/sec), 270

Figure 257: INCN HTTP Traffic Sent Houston (bytes/sec), 270

Figure 258: INCN HTTP Traffic Sent LA (bytes/sec), 270

Figure 259: HTTP Traffic Sent/Recvd Seattle, 271

Figure 260: HTTP Traffic Sent Chicago (bytes/sec), 271

Figure 261: INCN E-Mail Upload Response Time (sec), 273

Figure 262: INCN E-Mail Download Response Time (sec), 273

Figure 263: INCN E-Mail Traffic Sent (bytes/sec), 273

Figure 264: INCN E-Mail Traffic Received (packets/sec), 273

Figure 265: INCN E-Mail Traffic Sent NY (bytes/sec), 274

Figure 266: INCN E-Mail Traffic Sent/Received Miami (bytes/sec), 274

Figure 267: INCN E-Mail Traffic Sent LA (bytes/sec), 274

Figure 268: INCN E-Mail Traffic Sent Chicago (bytes/sec), 274

Figure 269: INCN TCP Delay (sec), 276

Figure 270: INCN IP Traffic Dropped (packets/sec), 277

Figure 271: INCN Ethernet Delay (sec), 278

Figure 272: INCN Link Statistics New York – Miami, 280

Figure 273: INCN Link Statistics Miami – Houston, 282

Figure 274: INCN Link Statistics Houston – Chicago, 284

Figure 275: INCN Link Statistics Houston – Los Angeles, 286

Figure 276: INCN Link Statistics Los Angeles – Seattle, 288

Figure 277: INCN Link Statistics Seattle – Chicago, 290

Figure 278: INCN Link Statistics Chicago – New York, 292

Figure 279: DB-Tech-Appl NY Server Traffic Sent (bytes/sec), 295

Figure 280: DB-Tech-Appl NY LAN-10 Traffic Sent (bytes/sec), 295

1. Introduction

If intellectual capital is like cash in the bank, then knowledge flow must be like cash flow. Intellectual capital is gradually being recognized as a true asset to an individual, though there are many theories on how to measure it. Like intellectual capital, knowledge flow is becoming recognized as a strong indicator of the individual's capacity to adapt, to innovate, and to survive. The fastest way for knowledge flows is found in a networked environment where anyone that can access it is able to exchange any type of information with anyone else at any time and any place. More specifically, in the educational area that promotes the generation and propagation of community intelligence, the integrative capability of a campus network can be visualized by means of a multi-layered pyramid, where the data processing chores are enriched with a real-time communications discipline. This concept is comprehensively presented in figure 1, where the bottom layer focuses on transactions and the top layer on strategic issues.

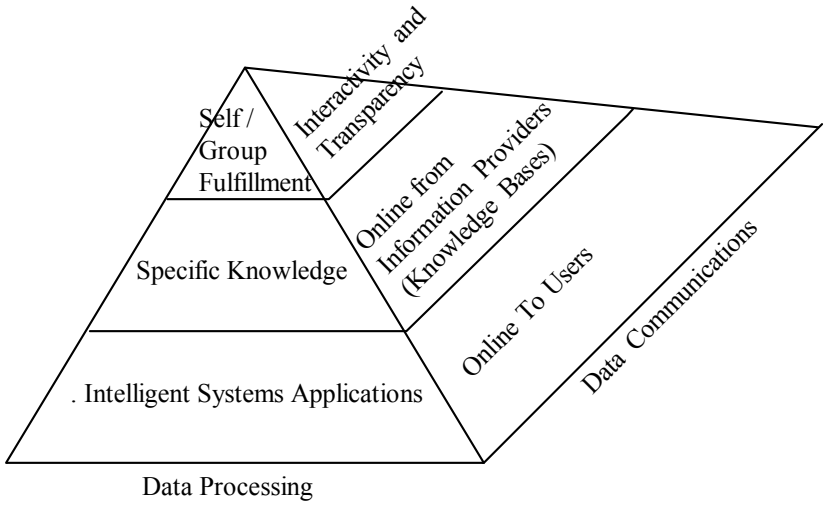


Figure 1: Generation & Propagation of Community Intelligence

The impact of global networks, linked through connections with a global reach, e.g. Internet, specifically in education is only now beginning to be felt on a global scale, due to the widespread availability of computer technology that would seem a science-fiction dream to most people for almost all of the twentieth century. The promises of multimedia, simulation, computer-mediated communication and communities, and Internet-based support for individual and distance learning all have the potential for revolutionary improvements in learning. Educational systems in this country and others are now struggling to achieve this revolution, learning how to best use these new technologies in their circumstances. [1]

This research aims to provide insights into the underlining technological network model for the implementation of what is being seen as the next revolution in education (the impact of the Internet in education has already been widely examined): the widespread use of optimally connected campus networks to facilitate the manipulation of knowledge located in repositories around the world. One example of such "revolution" is Internet 2, implemented via the Abilene network which provides connections between universities around the world [5].

The driver of this revolution has been scientific progress. This progress comes from two sources. Progress in artificial intelligence has led to a deeper understanding of how to present knowledge, how to reason, and how to describe "how to" knowledge itself, i.e., procedural knowledge. The second source is progress in cognitive science, which has led to a deeper understanding of how people think, solve problems, and learn.

It is irrefutable the synergy that exists between these two areas: artificial intelligence and cognitive science. Cognitive scientists often use artificial intelligence techniques to create

simulation models of cognitive processes and behaviors. Artificial intelligence scientists use results from cognitive science to guide their explorations, and to create software with more human-like abilities. When applied to education, this synergy can lead to combinations of software and hardware that allow for activities that can help more students to achieve better learning.

The question is how to establish means to make any valuable knowledge available to any knowledge seeker. Undoubtedly, interconnected networks can provide the superhighway for our education in any field of culture of anyone's choice. Internet is a proof of this. Three important tasks must be accomplished in order to extend network access to all knowledge seekers, and enable them to incorporate these resources and techniques into their everyday educational activities. First, schools and homes must be provided with connectivity, second, facilities must be developed that make it easy and inexpensive to create and access online resources, and third, the exchanges of information between the communication media and the user should result in a pleasant and efficient knowledge acquisition by the user. Indeed, we could expect that the ideal world would be the one where all nations are interconnected permitting them to communicate knowledge among themselves and establishing a commonality of basic life values that suppress ignorance and rebellious behaviors.

Of course, technical underpinnings are only part of the much larger issue of major educational reform. Human factors ultimately will determine the extent to which networking technology can assist the task of educational reform. But technology can significantly reduce the human effort that must go into educational reform and can ease the cost of the transitions associated with reforms. The real challenge is to use the

technology effectively to unlock the imaginations and abilities of students and teachers across the nation.

As universities and research institutions become more sensitive to their educational missions, a theoretical construct can take hold in which scientists and scholars understand that the fundamental purpose being served in the creation of knowledge is the advancement of education at all its levels, making it available at any time and anywhere.

There is a need for a design approach which results in a reliable communications network system that provides easy access to knowledge to anyone seeking it in any part of the world. An initial instance of this development is the tremendous academic energy surging into Web site development, with many projects that go way beyond facilitating research in the narrow sense to promoting broad participation in intellectual work. Advancing knowledge becomes a defining good and necessary goal of the human policy, and participation in that enterprise becomes both a feasible endeavor and a basic human right for all its citizens.

A lot of knowledge is out there residing in databases. In order to empower citizens to acquire whatever knowledge they want, the goal is to identify, design, and implement network applications that would provide an easy interaction with the user and which would augment its knowledge about the user as the means to supply the user with the best service that is being requested. It would result in the user's self-fulfilling satisfaction experimentation due to the familiarization of both parties: the user and the system. This is precisely the goal of this thesis. However, the broader ambition of the thesis is constrained by the formidable task of engineering all the parts of such system, and we are

at the end left with the satisfaction of accomplishing the initial task of at least uncovering some of the parts of the system.

The thesis is organized hierarchically. The basic concepts are followed by the implementations and conclusions. We first review the basic concepts of communication networks. Next we introduce characteristics of campus networks (CNs). Afterwards, we describe a range of current educational technologies used in these CNs, and how these technologies support knowledge processing. We then present our network design, which is going to be the model for each and every node in our national and international networks. Next simulation is run for different scenarios in order to analyze the data traffic under these scenarios. At this point we have garnered knowledge about optimization of campus networks and then we move to insert intelligence to the network in order to make it capable to handle queries intelligently. We make use of the concept of Intelligent Networks (IN) and blend it with the IP campus networks. The system is given the capability to recognize the knowledge domain the queries belong to according to the Dewey scheme and after packets identification, the system forwards the queries to the respective knowledge base that can be located anywhere in the world. This intelligent educational systems function as a ‘negotiator’ between a knowledge seeker and the knowledge provider.

Thus, our model of an optimized intelligent campus network with quality services and efficient routing mechanism for providing intelligent services to users can be the prototype of an intelligent tutoring system for the novices to the experts on any knowledge domain that can be accessed in any point of the world.

2. Communication Networks

A communication network is a set of equipment, facilities, and personnel that provides a service: the transfer of information between users located at various geographical points. Examples of such networks include telephone networks, computer networks, television broadcast networks, cellular telephone networks, and the Internet. The exchange of information in a network involve the use some mechanism that is able to switch the information flow from one transmission line to another.

Switched networks involve a partially or fully (any node is connected to any other node in the network) meshed topology and use special network devices called switches to interconnect the links between source and destination nodes.

In addition to geographical area and topology, switched networks can also be classified by the type of communications path they use and the manner in which data are transmitted across this path. Two particular classifications are:

. circuit-switched, and

. packet-switched.

Circuit-switched networks are also known as connection-oriented networks, while packet-switched networks are said to be connectionless. Also, circuit-switched networks require dedicated links for communication, while packet-switched networks use links that can be shared by more than one communication.

2.1 Circuit-Switched Networks

In a circuit-switched network, a dedicated physical circuit is first established between the source and the destination nodes before any data transmission takes place. For this reason, circuit-switched networks are also called connection-oriented networks. Furthermore, the circuit remains in place for the duration of a transmission.

The public switched telephone network (PSTN) is an example of a circuit-switched network. When we place a telephone call, a direct physical communications path is established between our telephone set and the receiver's telephone set. The set is a point-to-point connection that interconnects the telephone company's switches, which are located throughout the telephone network. Once established, the circuit is dedicated exclusively to the current transmission. After the transmission is completed, this dedicated circuit is then released and made available for another communication transmission. Thus, circuit-switching promotes link sharing since the same circuits can be used for different transmissions. Examples of connection-oriented networks: X.25, frame relay, and ATM. The following section reviews X.25, while frame relay and ATM are discussed later as one of the high speed broadband networks.

2.1.1 X.25

Although X.25 may be phasing out, it's worth studying as an example of a connection-oriented reliable network layer. X.25, which was designed by ITU-T, as its connection-oriented network layer, with the name ISO8208. X.25 does not specify how to build a connection-oriented service. Instead, it specifies the interface between an end node (known in X.25 as a DTE) and a router (known in X.25 as a DCE).

The service as seen the DTE (Data Terminal Equipment) involves the ability to carry on multiple simultaneous calls over a single link to a DCE (Data Circuit-terminating Equipment) as seen in the figure 2. Each call must first be established; then data can flow in both directions between the endpoints of the call, after which the call must be cleared.

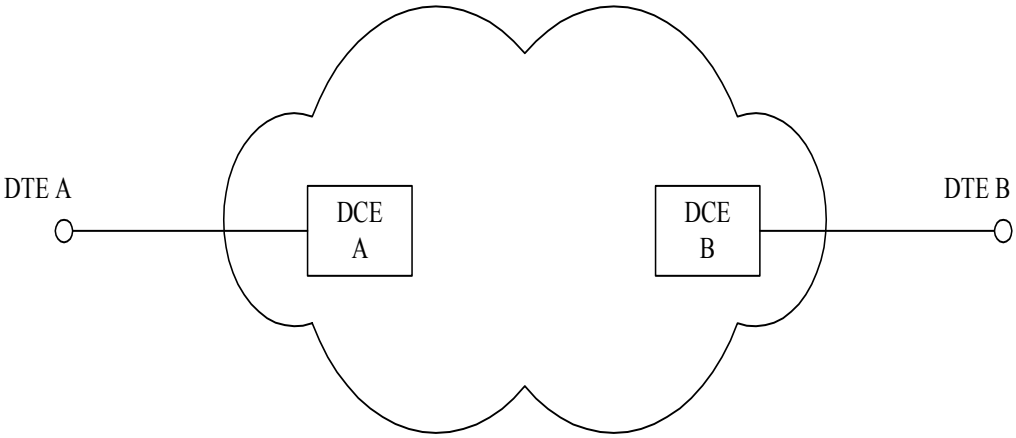


Figure 2: X.25 network

2.2 Packet-Switched Networks

In a packet-switched network, messages are first partitioned into smaller units called packets, which are then sent to the destination node one at a time via intermediate switches. A packet represents the smallest unit of data that can be transferred within a given network. Each packet carries the destination node's address as well as a sequence number. When a packet arrives at an intermediate switch, the switch examines the packet's destination address to determine which path the packet should take to the next switch. This switching technique in which data are stored on one node of a point-to-point link and then forwarded to the next node repeatedly en route to the destination node is called store-and-forward. The concept of store-and-forward transmission requires that the

entire contents of a transmitted message (or packet) be received by each intermediate node before it is forwarded to a succeeding node.

Packet-switched networks also promote link sharing by using two transport schemes:

. Virtual Circuit

. Datagram

In virtual circuit packet-switching, instead of using a dedicated, physical circuit for every node-to-node communication, nodes share a communications channel via a virtual circuit. A virtual circuit is a logical communications path instead of a physical one. That is, it is a non-dedicated logical connection through a shared medium that gives the high-level user the appearance of a dedicated, direct physical connection from the source node to the destination node. The individual packets follow the same communications path in sequence as if they were traveling along a dedicated circuit. The difference is that links within this circuit can also be used for other transmissions at the same time, like cars in different lanes sharing the same highway.

In datagram packet-switching, packets are transmitted independently of one another at any time. Thus, it is possible for different packets form the same message to be transported across different communications paths. Furthermore, the packets are not necessarily transmitted in a specific order, which implies that the specified destination node is responsible for reassembling them in the correct order. (This is why packets also carry sequence numbers.) Most modern-day computer networks, including the Internet, are packet-switched [2].

3. High-Speed Broadband Technology

Network designers are faced with increasingly amount of network traffic that is causing slow-downs and bottlenecks. Many of them report an average of 15 to 20 percent network-growth over the course of a year [32]. This traffic is identified as mostly IP traffic and one of the short-term solutions to this problem is to increase bandwidth by replacing the existing circuits with high-speed broadband technology. Broadband technology is capable of offering high bandwidth and it can accommodate high-load multimedia data traffic applications. Until recently ISDN was the most common network technology for multimedia transmission. However it is gradually being replaced by more efficient technology such as x-DSL, SMDS, Frame Relay, SONET, and ATM.[3]. These technologies can be used in LAN-LAN, LAN-WAN, and WAN-WAN interconnections.

3.1 ISDN (Integrated Services Digital Network)

ISDN is a network designed to carry sound, images, data files and video over its network. It uses a double twisted pair cable but instead of using AC and DC voltages to provide the signaling protocol, the signaling is performed digitally by sending and receiving messages down a special channel – the D channel.

Thanks to the introduction of the digital network, we can now have 2B+D channels in a subscriber line. By 2B we mean two baseband channels; D means one data channel. By using the D channel we can transmit information to identify the ID of the source phone number. So when the phone rings the source ID can be show on the display of the telephone before we answer the call. This gives us a choice of whether or not to answer

the call. ISDN was posed to become widely used as line tariffs and equipment reduce in cost and availability and interoperability improves. However, as it is, a narrowband digital network (N-ISDN), its slow introduction and rampant consumption of bandwidth within many multimedia applications has provided support to the change to faster networks such as B-ISDN, xDSL, and ATM. B-ISDN, broadband ISDN, provides the high-speed transmission of digital data through the networks. B-ISDN provides interactive services and a distribution services. The interactive supports conversation, message handling, and information retrieval services. The distribution allows broadcasting services, such as radio or a television services. B-ISDN was introduced in late 1980s and its goal is to convert existing ISDN services to run over ATM, and it was generally based on the familiar narrowband ISDN standard.

3.2 xDSL (x Digital Subscriber Line)

The x stands for the DSL family: HDSL (High Bit-Rate DSL), ADSL (Asymmetrical DSL), IDSL (ISDN DSL), SDSL (Symmetrical DSL), RADSL (Rate Adaptive DSL), and VDSL (Very High Bit-Rate DSL). DSL is a technology that allows existing twisted pair telephone lines to transmit multimedia materials and high-speed data. Transfer speeds range from hundreds of thousands of bits per second to up to several million bits per second. The transfer speed of a particular line depends on one or more factors: the carrier providing the service, the distance of your house or business from the central office of the local telephone company, and whether the DSL service is a symmetric connection or an asymmetric connection. The symmetry relates to the speeds for uplinking and downlinking to the service provider.

3.3 SMDS (Switched Multi-megabit Data Service)

SMDS is a cell-based, connectionless, high-speed, public, packet-switched, broadband, metropolitan area data network. SMDS is also a data service, which means that it can only transmit data (although, it is capable of transmitting voice and video).

SMDS uses fixed-length cells as its basic units for transmitting data. These cells are similar to ATM cells, namely, they contain 53 bytes – a 44-byte payload plus a 7-byte header and a 2-byte trailer. For a data transmission, SMDS switches establish a virtual circuit, instead of a dedicated one, between the sending and receiving machines. When they are ready, data cells are sent independently of each other without delay and in no particular order. This provides bandwidth on demand for the kind of bursty traffic inherent in LAN applications. In contrast, ATM and frame relay are connection-oriented.

3.3.1 SMDS Cell Structure

When SMDS data cells are received by a SMDS node (router/switch), this node encapsulates these data cells into SMDS data packet(s) which are then sent over the link. The packet format is shown in figure 3. It has three fields: the destination (where the packet is to go to), the source (who sent it), and a variable length payload field for up to 9,188 bytes of user data.

Bytes	8	8	≤ 9,188
	Destination address	Source address	User data

Figure 3: SMDS Cell Structure

3.3.2 SMDS Operation

The machine on the sending LAN that is connected to the access line puts the packet on the line and SMDS makes a best effort attempt to deliver it to the correct destination. No guarantee is given that the packet will be received by the intended destination machine. When a packet arrives at the SMDS network, the first router checks to make sure that the source address corresponds to the incoming line, to prevent billing fraud. If the address is incorrect, the packet is simply discarded, otherwise the packet is sent to its destination.

A useful SMDS feature is broadcasting. The customer can specify a list of SMDS telephone numbers, and be assigned a special number for the whole list. Any packet sent to that number is delivered to all members on that list. An additional user feature is address screening, on both outgoing and incoming packets. Customers, for example, can give a list of telephone numbers and specify that no packets may be sent/received to/from any of these numbers. The payload can contain any byte sequence the user wishes, up to 9,188 bytes. SMDS does not look at it. It can contain Ethernet packet, token ring packet, IP packet, or anything else.

3.3.3 SMDS Speeds

SMDS can support several data rates including DS-1 (1.44 Mbps), DS-3 (45 Mbps), and SONET OC-3 (155 Mbps). Low-speed SMDS (56 Kbps/64 Kbps) is also available from some telcos. Since SMDS is a technology-independent service, customers can subscribe to SMDS at a specific data rate for guaranteed throughput.

3.4 Frame Relay

Frame Relay is a public WAN packet-switching protocol that provides LAN-to-LAN connectivity. Its name implies what it does, namely, relays frames across a network between sites. What frame relay provides is a single connection into a public network instead of multiple interconnections. However, this single connection is not a dedicated line but rather a virtual link. Thus frame relay is a connection-oriented protocol that requires that first a connection must be established before two network nodes can communicate. Frame relay relies on permanent virtual circuits (PVCs) to interconnect two sites. Frame relay is capable of supporting services such as on-demand connections, multicast communications, data compression, and CoS/QoS, all of which enable it to transmit multimedia data such as voice and video.

Frame relay frames can be encapsulated within ATM's 53-byte cell and transmitted across ATM links. Furthermore, given frame relay's variable frame size, a single frame can carry a minimum-sized Ethernet/802.3 frame (64 bytes), whereas ATM needs two cells for the same transmission.

3.4.1 Frame Relay Frames

The frame relay frame shown in the figure 4 is of variable length, the standard does not specify a frame size limitation. The maximum size is set (and is different among) the service providers and is generally a function of hardware limitations. Typical implementations range from 1,600 to 4,096 bytes. This implies that frame relay can fully support Ethernet/802.3 and token ring frames without having to segment (and reassemble) LAN frames for transmission across the frame relay network.

Header		Payload	Trailer	
SFD	Address	User Data	FCS	EFD
Start Frame Delimiter (1 byte)	2bytes	Variable number of bytes	Frame Check Sequence (2 bytes)	End Frame Delimiter (1 byte)

Figure 4: Frame Relay Frame

3.4.2 Frame Relay Operation

Frame relay is a data link layer protocol, which means that frame relay network nodes will only be concerned with the physical and data link layers functions. This implies in physical characteristics of the link plus addressing.

Frame Relay is synchronous in nature and is based on the concept of packet-switching. Thus, every frame relay frame carries source and destination physical addresses.

It uses statistical multiplexing. This enables multiple subscribers to share the same backbone. Frame relay switch buffers that get filled will discard any incoming packets. However frame relay nodes do not provide for flow control, error detection, frame sequencing, or acknowledgements. Recovery from lost/discarded frames is left to the frame relay end nodes (usually a router, at the customer's site or at the central office of the public carrier). Collectively, end nodes are known as frame relay access devices (FRADs). The path nodes, frame relay switches, ignore transmission errors.

Frame relay is protocol independent, i.e., different LAN protocols such as Ethernet, Token Ring, TCP/IP, SNA, and IPX can run over frame relay.

3.4.3. Frame Relay Speed

Frame relay, which originally was a subset of ISDN, was defined initially to transmit data at rates ranging from 56 Kbps to 1.44 Mbps (T-1). It was considered an interim technology, lying somewhere between X.25 and ATM relative to their respective data transmission rates. In North America, frame relay has pushed SMDS and B-ISDN out of the spotlight. Frame relay operates at T-3 (45 Mbps), and telcos are upgrading their networks to support even greater rates than DS-3. In fact, advances in technology is already implementing new frame relay recommendations for data rates up to OC-12 (622 Mbps), making frame relay competitive with ATM for WAN connections.

3.5 SONET (Synchronous Optical Network)

Synchronous Digital Hierarchy (SDH) provides welcome relief from the growing bandwidth problem in the WAN environment. SDH, see figure 5, introduced in the middle of 1980s, is a high bit rate fiber optic based transport method that provides the foundation for linking high-speed ATM switches and multiplexers, and provides users with B-ISDN compliant services.

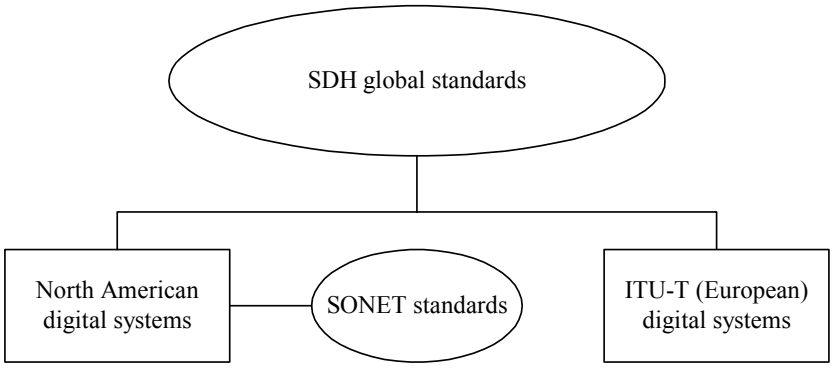


Figure 5: SDH Global Standards

SDH's proposal is based on the following assumptions:

- Fiber optic media are the transmission media of choice
- Network users require high-bandwidth services (broadband services)
- Remote switches connected by fiber links are highly desirable
- Switching equipment will accommodate a direct optical interface

SONET is the North American version of SDH, the international standard methodology for a family of high-speed transmission links on fiber optic cable.

SONET defines a hierarchical set of transmission rates and transmission formats. These standards are to be used by equipment vendors and service providers at or above a basic optical transmission rate of 51.840 Mbps, which forms the base transmission rate for SONET. Today SONET is capable of transmission speeds as high as 10 Gbps.

3.5.1 SONET Frame

The basic building block of the SONET digital transmission hierarchy is called the STS level 1 (STS-1) which consists of 810 bytes transmitted 8,000 times per second, or once every 125μs, to form a 51.84 Mbps signal rate. The basic SONET frame, figure 6, consists of a rectangular array of bytes arranged in 9 rows by 90 bytes is repeated 8000 times a second. Thus each byte in the array corresponds to a bit rate of 64 Kbps, and the overall bit rate of the STS-1 is 51.84 Mbps. The STS-1 frame is transmitted one row at a time, from top to bottom, and from left to right within each row. Therefore the byte B in row 1, column 1 is sent first, and the byte B in row 9, column 90 is sent last.

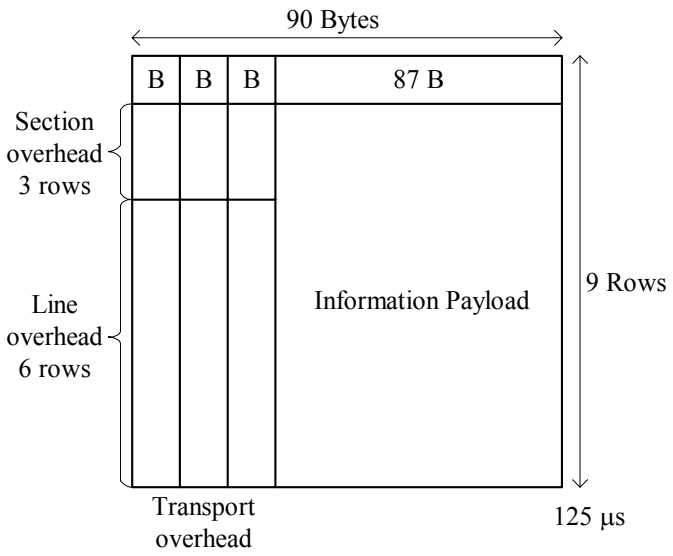


Figure 6: SONET frame

One frame is sent every 125 ms; thus SONET can maintain time-slot synchronization that is required for delivery of normal, uncompressed voice modulation (Pulse Code Modulation, PCM, method), which is 8 bits 8,000 times per second or 64 Kbps.

3.5.2 SONET Speeds

The SONET standard defines a transmission rate hierarchy that begins where the current digital hierarchy ends. In North America, the digital hierarchy ends with the T-carrier systems, whose rate is 45 Mbps (T3 speed); in Europe's ETSI systems, that rate is 139 Mbps. The SONET hierarchy, shown in the table 1, consists of a few basic building blocks of terms and speeds. Each level has an optical carrier (OC) level and an electrical level called the synchronous transport signal (STS). For example, an STS-3 frame is sent on an OC-3 fiber link, just as a T3 frame is sent on a T3 link in the T-carrier hierarchy. The counting units are the basic 51.84 Mbps rate of OC-1.

Optical Level	Electrical Level	Line Rate (Mbps)	SDH Equivalent
OC-1	STS-1	51.84	_____
OC-3	STS-3	155.520	STM-1
OC-9	STS-9	466.56	STM-3
OC-12	STS-12	622.08	STM-4
OC-18	STS-18	933.12	STM-6
OC-24	STS-24	1,244.16	STM-8
OC-36	STS-36	1,866.24	STM-13
OC-48	STS-48	2,488.32	STM-16
OC-96	STS-96	4,976.64	STM-32
OC-192	STS-192	9,953.28	STM-64

Table 1: SONET Speed Hierarchy

A unique difference between the SONET hierarchy and the T-carrier hierarchy is that in the SONET hierarchy the levels are related by a simple multiplication operation between them, for example, STS-3 operates three times faster than STS-1.

3.5.3 SONET Multiplexing

SONET employs a synchronous, byte-interleaved form for its multiplexing system. This means lower bit-rate channels can be identified, extracted, or inserted by a single device without the need to de-multiplex the higher SONET rate signal. This allows simplified

add/drop capability to insert and extract multiplexed streams of data. SONET uses the term tributary to refer to the component streams that are multiplexed together.

Figure 7 shows how a SONET multiplexer can handle a wide range of tributary types. A slow-speed mapping function allows DS1, DS2, and an E-1 (European similar to a North-American T1) signals to be combined into an STS-1 signal. Also shown is a DS3 signal being mapped into an STS-1 signal, an E-4 signal mapped into an STS-3 signal, and ATM streams mapped into an STS-3 signal. A SONET multiplexer (MUX) can then combine STS input signals into a higher-order STS-n signal which goes thru a scrambler and, in case of an optical link, the STS-n signal is converted to an OC-n signal.

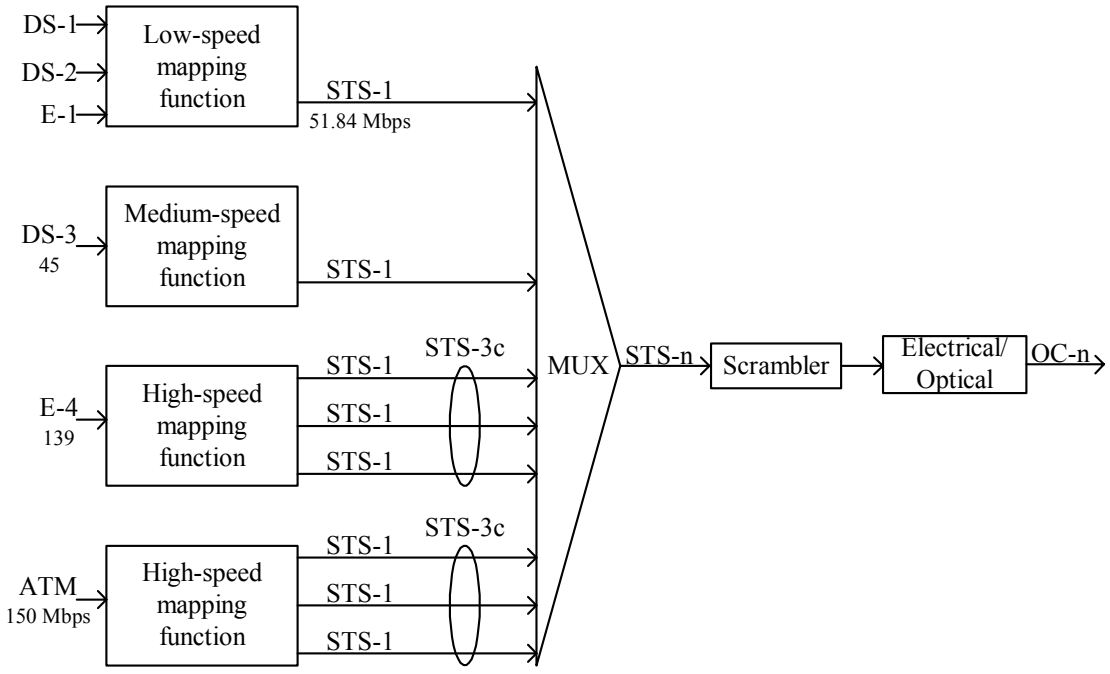


Figure 7: SONET Multiplexer Operation

SONET produced significant reduction in cost by enabling add-drop multiplexers (ADM) to add and drop tributary streams without disturbing tributary streams that are in transit.

3.5.4 SONET Advantages

It has become the de facto standard for the transmission of ATM cells that carriers bundle these cells into a SONET frame.

Table 2 shows the reasons for the use of SONET as a fast and reliable means of transport for optical signals.

Advantage	**Reasons**
Technology	Unprecedented speeds available on fiber optic cable
Economics	Best interface for fiber optic networks; Economical adding and dropping of channels
Flexibility	Modular equipment design; Adequate overhead for network management
Compatibility	Works well with existing network hierarchies; Allows multiple vendors' equipment to interoperate; Worldwide standard (Synchronous Digital Hierarchy)

Table 2: Advantages of Using SONET

SONET is the most deployed WAN link technology, but it also has its disadvantages, chief among them is the fact that SONET is a multiplexing scheme and packets need to be framed one link protocol over the next, e.g. packets over SONET is layered as IP over PPP over SONET, making SONET transceivers very complex.

3.6 Asynchronous Transfer Mode (ATM) Networks

ATM is a project invented in the mid 1980s by the telephone industry, to provide a capability to deliver information anywhere at speeds that already exceeds Gigabits per second. ATM is basically a high-speed packet-switching technology, and the telephone companies (telcos) have little experience with it. What they do have is a massive investment in a different technology (circuit switching) that is in concept unchanged since the days of Alexander Graham Bell. What these telcos are realizing, it is the possibility to market ATM technology to organizations that have run out of bandwidth on their campus or building-wide LANs and are being forced to go to some kind of switched system that has more bandwidth than does a single LAN, or to institutions that their client-server systems need the ability to talk to certain servers at high speed.

3.6.1 What exactly is an ATM?

The basic principles of ATM as put forward by CCITT in Recommendation I.150 are:

- ATM is a specific packet oriented transfer mode based on fixed length cells.
- ATM is connection-oriented, thus it provides greater control of network resources.
- The information field of ATM cells is carried transparently through the network.
- Contrary to IP which supports data, voice, and fax, ATM can transport all services (data, voice, and video) including connectionless services., see figure 8.

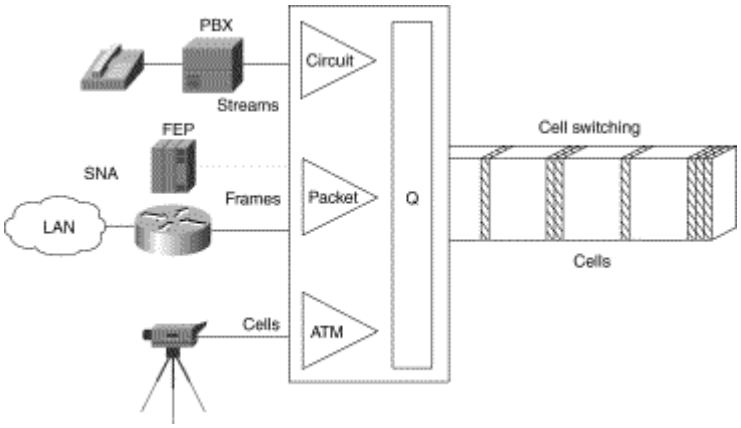


Figure 8: ATM support of various traffic types.

ATM utilizes fixed-length packets to carry different types of traffic at different speeds, and enable carriers to capitalize on a number of revenue opportunities through multiple

3.6.2 How does it Work?

ATM divides voice, video, and data signals into packets, or cells, that can be intermingled and sped along a single line. The cells are all the same size, 53 bytes, so multiple cells can be handled quickly and routed simultaneously.

Voice communications require relatively little transmission capacity, or bandwidth, to move through a network. Video signals, on the other hand, require a very broad bandwidth to go from point to point. Figure 9 shows the cell format.

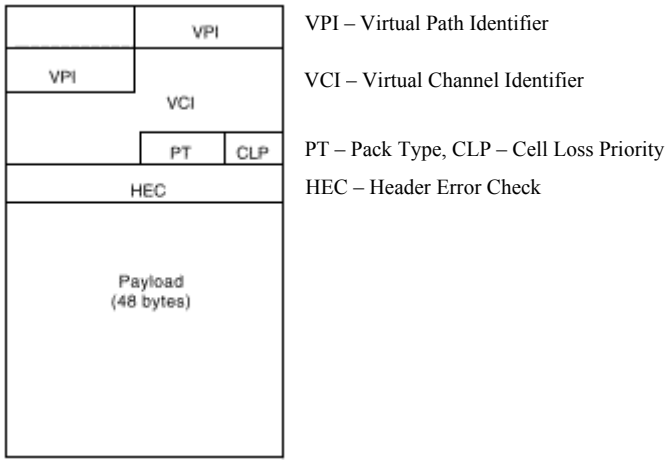


Figure 9: ATM Cell Format

ATM networks are connection-oriented. Setting up an ATM connection, figure 10, requires first sending a message to set up the connection. After that, all subsequent cells follow the same path to the destination. Cell delivery is not guaranteed, but their order is. If cells 1 and 2 are sent in that order, then if both arrive, they will arrive in that order.

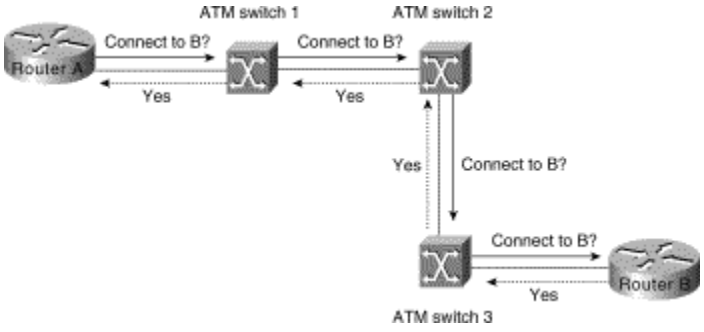


Figure 10: Establishing a connection in an ATM network

\An ATM connection must be established between two end points before any data transfer can occur. This connection is accomplished through a signaling protocol as shown below:

For Router A to connect to Router B the following must occur:

1. Router A sends a signaling request packet to ATM Switch 1. This request contains the ATM address of the Router B as well as any QoS parameters required for the connection.

2. ATM Switch 1 reassembles and examines the signaling packet from Router A.

3. If ATM Switch 1 has an entry for Router B's ATM address in its switch table and it can accommodate the QoS requested for the connection, it sets up the virtual connection and forwards the request to the next switch ATM Switch 2.

4. Every switch along the path to Router B reassembles and examines the signaling packet, and then forwards it to the next switch if the QoS parameters can be supported. Each switch also sets up the virtual connection as the signaling packet is forwarded. If any switch along the path cannot accommodate the requested QoS parameters, the request is rejected and a rejection message is sent back to Router A.

5. When the signaling packet arrives at Router B, Router B reassembles it and evaluates the packet. If Router B can support the requested QoS, it responds with an accept message. As the accept message is propagated back to Router A, the switches set up a virtual circuit.

6. Router A receives the accept message from its directly connected ATM switch (ATM Switch 1), as well as the Virtual path identifier (VPI) and Virtual channel identifier (VCI) values that it should use for cells sent to Router B.

ATM circuits are of two types, figure 11:

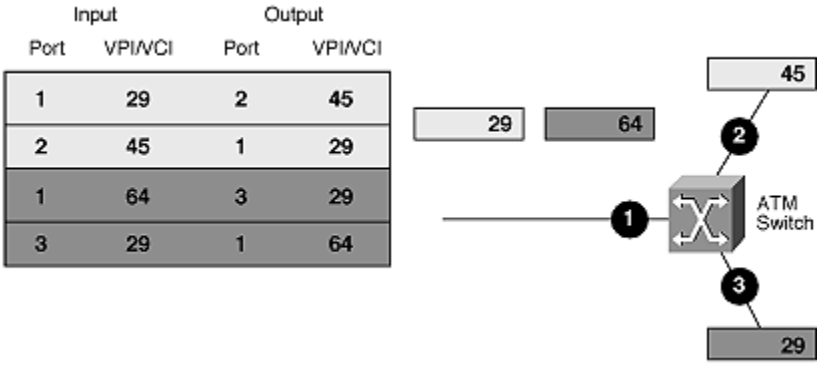
. virtual paths, identified by virtual path identifiers (VPI), and

. virtual channels, identified by a VPI and a virtual channel identifier (VCI).

A virtual path is a bundle of virtual channels, all of which are switched transparently across the ATM network on the basis of the common VPI.

Figure 11: ATM Virtual Paths and Virtual Channels

The basic operation, figure 12, of an ATM switch is very simple: to receive a cell across a link on a known VCI or VPI value it looks up the connection value in a local translation table to determine the outgoing port (or ports) of the connection and the new VPI/VCI value of the connection on that link, and then retransmits the cell on that outgoing link with the appropriate connection identifiers.



Input		Output	
Port	VPI/VCI	Port	VPI/VCI
1	29	2	45
2	45	1	29
1	64	3	29
3	29	1	64

Figure 12: ATM Switch Operations

The switch operation is so simple because external mechanisms set up the local translation tables prior to the transmittal of any data. The manner in which these tables are set up determine the two fundamental types of ATM connections, see figure 13:

- *Permanent Virtual Connections (PVC):* PVC is a connection set up by some external mechanism, typically network management, in which a set of ATM switches are programmed with the appropriate VPI/VCI values.

- *Switched Virtual Connections (SVC):* An SVC is a connection that is set up automatically through a signaling protocol. SVCs do not require the manual interaction needed to set up PVCs. All higher layer protocols operating over ATM primarily use SVCs, and it is these that are primarily considered in this paper.

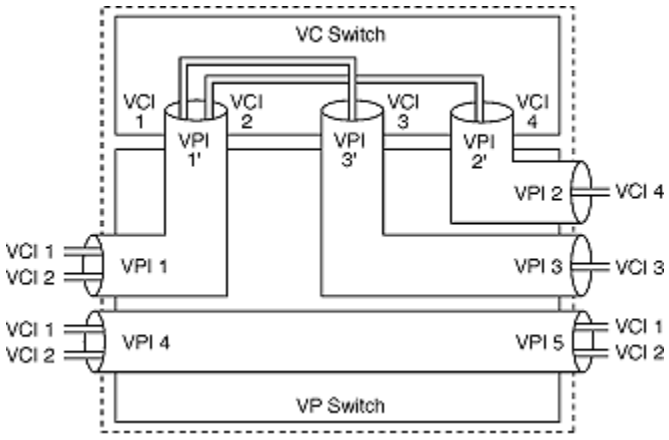


Figure 13: Virtual Circuit and Virtual Path Switching

3.6.3 How fast is it?

The intended speeds for ATM networks are 155 Mbps and 622 Mbps, although speeds in the Gigabits per second range are now being achieved. The 155 Mbps speed was chosen because this is about what is needed to transmit high definition television. Table 3 lists different high speed wide area connection technologies.

	Speed (Mbps)	Switchable/Fixed	Scalable	QoS	Availability
N-ISDN	0.128	Both	No	No	Wide
DSL	0.64 – 7	Fixed	Yes	No	Growing
ATM	1.5 – 622	Fixed	Yes	Yes	Wide

Table 3: High-Speed Wide Area Connection Technologies

3.6.4 ATM Model

ATM model consists of three layers, figure 14, the physical, ATM, and ATM Adaptation layers, plus whatever the users want to put on top of that.

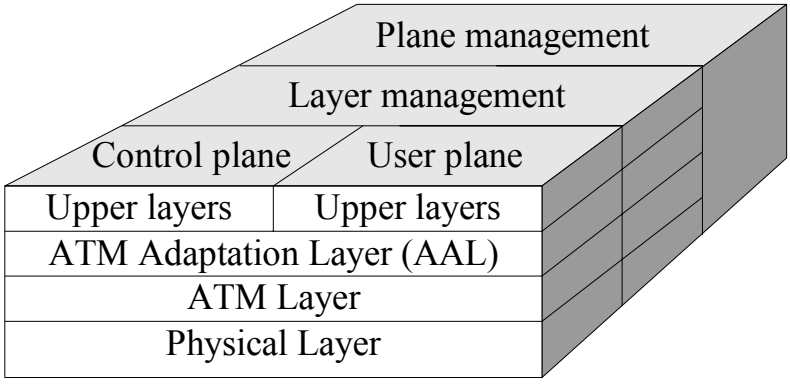


Figure 14: ATM Model

Physical Layer: defines the transmission medium, bit transmission, encoding, and electrical to optical transformation. ATM Layer: provides routing, traffic management, switching, and multiplexing services. ATM Adaptation Layer (AAL): accepts transmissions from upper layers (packet data) and map them into fixed-size ATM cells.

3.6.5 ATM Structure

ATM is based on the concept of two end-point devices communicating by means of intermediate switches. As the figure 15 shows, an ATM network is made up of a series of switches and end-point devices. The end-point devices can be ATM-attached end stations, ATM-attached servers, or ATM-attached routers.

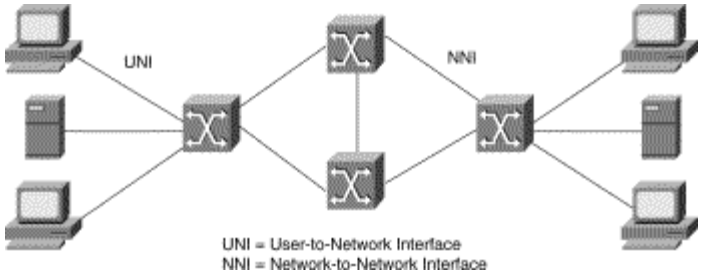


Figure 15: ATM Network

There are two types of interfaces in an ATM network: (i) User-to-Network Interface (UNI) , and (ii) Network-to-Network Interface (NNI). UNI is made up of an end-point device and a private or public ATM switch. NNI is the connection between two ATM switches. The UNI and NNI connections can be carried by different physical connections.

3.6.6 How ATM networks prioritize traffic

ATM uses a technique known as Class of Service to prioritize traffic, which is performed on the ATM switch. The quality of the service (QoS) is determined by two main parameters: cell delay and cell loss. Cell delay specifies the maximum amount of time that can elapse between when the cell is sent and when it is received. Cell loss refers to the number of cells that can be dropped due to congestion.

ATM combines four Class of Service:

- Constant Bit Rate (CBR) – minimum cell delay and some cell loss: real-time voice and video applications
- Variable Bit Rate (VBR) – low cell loss and low cell delay: real-time applications, but tolerate high cell delay for non-real-time applications: bursty data transfer. Available Bit Rate (ABR) – low cell loss, high cell delay: LAN data traffic
- Unspecified Bit Rate (UBR) – high cell loss and high cell delay: public WANs

3.6.7 ATM Cell Switching and Multiplexing

Presently, the variety of packet sizes makes network traffic unpredictable. Switches, multiplexers, and routers must incorporate elaborate software systems to manage the various sizes of packets. A great deal of header information must be read, and each bit counted and evaluated to ensure the integrity of the every packet. However, problem arises when the network needs to multiplex packets of different size and with different transmission requirement onto one link. Adopting a concept called cell networking solves many of the problems associated with packet internetworking. This technology is basically the ATM technology.

ATM uses asynchronous time-division multiplexing (ATDM) to multiplex cells coming from different channels. It uses fixed-size slots (size of a cell). ATM multiplexers fill a slot with a cell from any input channel that has a cell; the slot is empty if none of the channels has a cell to send. Switching and multiplexing can be implemented in hardware rather than software, resulting in less expensive devices.

3.6.8 ATM and SONET

ATM, like other protocols such as 802.5 Token Ring and Ethernet, can run over several different physical media and physical-layer protocols. The frequent ways of carrying ATM cells is inside a SONET frame, although SONET can also carry variable-length packages, such as FDDI, SMDS, and ISDN, while being able to carry fixed size ATM cells. SONET can also support various topologies including point-to-point, star, and ring. When you send ATM cells over some physical medium, the main issue is how to find the boundaries of the ATM cells. With SONET, there are two easy ways to find the boundaries. The first technique is that the overhead bytes in the SONET frame can be used as a pointer into the SONET payload to the start of the ATM cell. Having found the start of one cell, it is known that the next cell starts 53 bytes further on in the SONET payload, and so on. The other way to find the boundaries of ATM cells takes advantage of the fact that every cell has a CRC in the fifth byte of the cell. Thus, if you run a CRC calculation over the last 5 bytes received and the answer comes out to indicate no errors, then it is probably true that you have just read an ATM header. If this happens several times in a row at 53 byte-intervals, you can be pretty sure you have found the cell boundary.

4. Campus Networks

A campus network can be thought of as multiple workgroups that can be logically separated from each other according to some administrative requirement specified by the network administrator. In a corporation, these workgroups can be divided into functional units such as marketing, engineering, finance, and so on, or by geographical boundaries such as a floor or a wiring closet. Each functional unit or workgroup has its own unique applications and services, which are commonly provided by their own servers. Therefore, Engineering has a certain set of users with their own needs for a file server. Network access to these local servers needs to be very fast and without any bottlenecks since any problems accessing these services adversely impacts the day-to-day functioning of these workgroups.

Let us step back a bit a look how the campus networks were and are established, maintained, and enhanced. As networking technology developed, universities' departments began to build computer labs and connect mini-computers, main frames and PCs to networks. These LANs were developed as departments could afford them with no campus-wide plan or standards. This was generally done with one time funding, part-time staff and help from central services agencies. Some university colleges and departments fund their networks much the same way today. The quality of network services varies widely depending on the colleges and departments financial resources and their commitment to IT.

The need for a backbone network to tie campus networks together was soon recognized. This was done for a short time on a coaxial network originally designed for cable TV. The universities, then, deployed a state-of-the-art fiber optic backbone. The backbone eventually connected college and department LANs together with an external connection to the Internet. Many were pleased with the new backbone, but some felt it improper that revenues from telephone bills were used to build the backbone. Colleges and departments starved for the lack of supply money from the legislature welcomed any reduction in charges. The result was rate reduction on phone bills with no revenue stream or funding model beyond minimal daily operation.

One of the earliest interconnected campus networks was BITNET (Because It's Time Network), whose goal was to create an inexpensive mechanism for universities to communicate by using electronic mail and exchange information using file transfer. BITNET was started in 1981 by CUNY (City University of New York) and Yale University. At its peak it connected about 3,000 mainframes at universities in the USA, Canada, Europe, South America, Asia, and Australia. By 1990, it had been largely replaced by NSFNET (National Science Foundation Network).

Campus networks typically use high-speed circuits to interconnect a series of LANs and provide connections to other backbones, MANs, WANs, and the Internet. The generalization is that a backbone that connects many backbones spanning several buildings at a single location is often called a campus network, see figure 17.

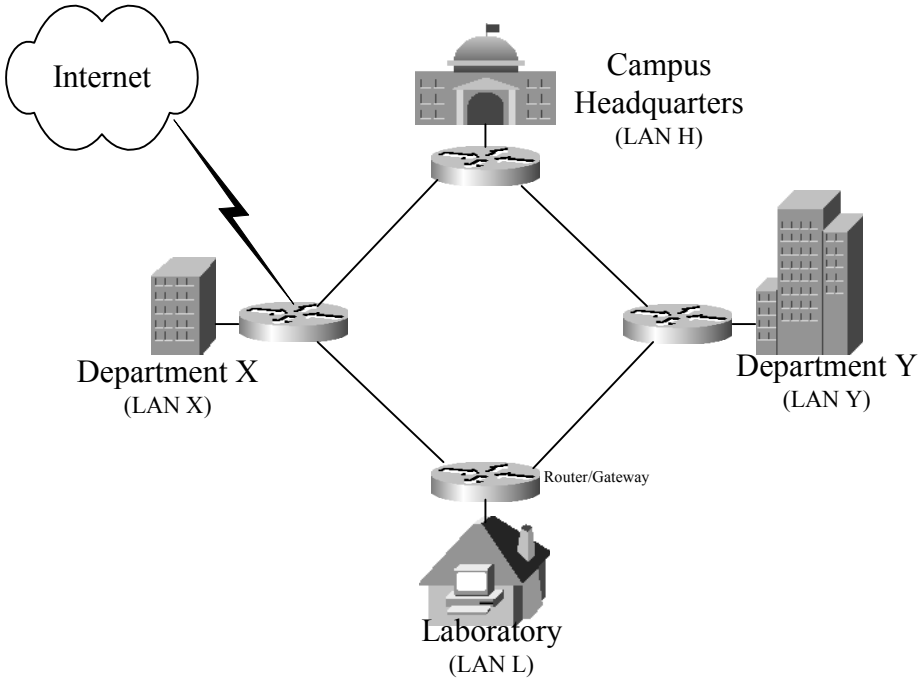


Figure 16: Typical Campus Network

This model is hierarchical, in that, multiple workgroups are interconnected using routers, while the basic functional unit, which is the workgroup network (usually switch-based network), is replicated multiple times.

The campus network topology is primarily LAN technology connecting all the end systems within the building. Campus networks generally use LAN technologies, such as Ethernet, Token Ring, Fiber Distributed Data Interface (FDDI), Fast Ethernet, Gigabit Ethernet, and Asynchronous Transfer Mode (ATM). A large campus with groups of buildings can also use WAN technology to connect the buildings.

The distinct characteristic of a campus environment is that the company that owns the campus network usually owns the physical wires deployed in the campus. After the wires are installed, bandwidth is inexpensive because the company owns the wires and there is

no recurring cost to a service provider. However, upgrading the physical wiring can be expensive. Consequently, network designers generally deploy a campus design that is optimized for the fastest functional architecture that runs on existing physical wire. For example, higher-speed technologies, such as Fast Ethernet, Gigabit Ethernet, and ATM as a backbone architecture, and Layer 2 switching (it uses the layer-2 data link layer address) provide dedicated bandwidth to the desktop. Network designers often think about three distinct technology layers when they design campus networks: access layer, distribution layer, and core layer, figure 17.

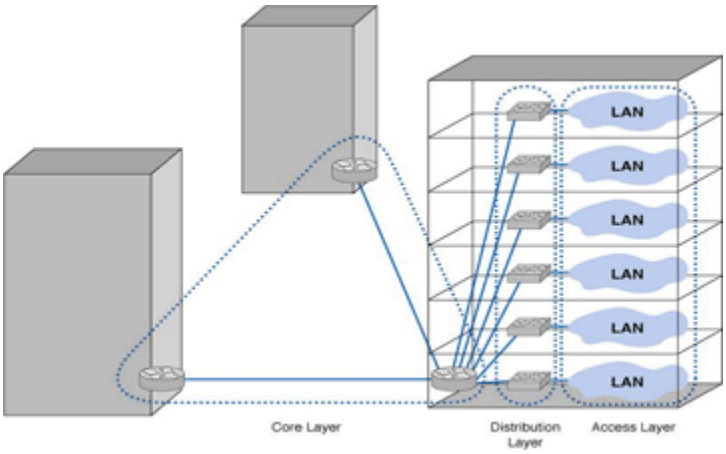


Figure 17: Layer Design of a Network

. Access layer – technology used in LANs

. Distribution layer - part of the backbone that connects the LANs together

. Core layer - connects different backbone networks together, often between buildings.

We will also make use of this approach in our model of a campus network: build up the design from the core layer to the distribution layer and finally to the access layer.

4.1 Campus Networks Architectures

There are four fundamental architectures that can be combined in different ways to design a campus networks: hierarchical backbone, flat backbone, collapsed backbone, and virtual LAN. These four architectures are mixed and matched to build sets of backbones.

a. Hierarchical Campus Networks:

- •Each LAN is a separate and isolated network, connected by a TCP/IP gateway (usually a router) to a shared media backbone network.
- •Within the LANs messages are sent based on the data link layer addresses.
- •To move between LANs, message traffic needs to be sent specifically to the router, which forwards the message based on its network layer address.

b. Flat Backbone:

- •With a flat backbone, LANs are connected using bridges or layer-2 switches.
- •Packets are forwarded based on their data link layer addresses, making the entire flat backbone a single subnet.
- •Flat backbones using bridges were developed in the mid-1980s to reduce costs, because at the time routers were very expensive.
- •Bridges have now become obsolete and are typically replaced by layer-2 switches, which have continued to fall in price.

c. Collapsed Backbone:

- • Uses a star topology, usually with a high speed switch at the center.
- •Collapsed backbones can use either layer-2 switches or layer-3 routing switches.

•The two main advantages are:

. each connection to the switch becomes a separate point-to-point circuit also giving much higher performance (from 200-600% higher)

. the network has far fewer devices and so is much simpler to manage.

•Two minor disadvantages are:

. use more cable and the cable runs for longer distances,

. if the central switch fails, the network goes down.

d. VLAN:

•VLANs, a new type of LAN/BN architecture using intelligent, high-speed switches.

•Unlike other LAN types, which physically connect computers to LAN segments, VLANs assign computers to LAN segments by software.

4.2 Access Methods in Campus Networks

The access methods to connect campus networks to other backbones networks can either be a simple point-to-point connection or via a fast internetwork which then is connected to the main backbone. The access networks can be any of the following: OC-n/STS-n (n = 1, 3, 9, 12, 18, 24, 36, 48, 96, 192 – see table 1), PPP, x-DSL, ISDN, Frame Relay (FR), SONET, ATM, and Wireless, or a combination of these broadband access technologies. The most common methods of transporting multimedia data to a customer premises are based on cable modem, digital subscriber line, and two wireless media methods (satellite and cell technology).

Our simulation of campus networks will only include full-duplex point-to-point links (T's and OC's). It is worth noting that the PPP (Point-to-Point Protocol) for OC links, SONET PPPs, can also be used to carry ATM cells. Knowing that this thesis includes OC-3 and OC-12 which are the standards ATM speeds of 155 Mbps and 622 Mbps and also the inclusion of quality of service, it is correct to say that our study, despite not including ATM switches, is a good indicative of how ATM network would behave under our scenarios.

4.2.1 Point-to-Point Links

Point-to-point links connect exactly two network nodes. Circuit-switched traffic can only be routed over point-to-point link groups, whereas packet-switched traffic can be routed over any of the link types. The capacity of a point-to-point link is the number of links for circuit switching or the total bandwidth (Mbps) of the group for packet switching.
Our simulation scenarios consist of similar nodes, all with the same network model, being interconnected through the following point-to-point full duplex links:

- T-3: a high-speed digital transmission line that is obtained by combining 28 DS1 signals. The DS3 signal has a speed of 44.736 Mbps
- OC-3: light sources that operate at a rate of 155 Mbps, which is the STS-3 rate. As mentioned before, STS refers to the electrical transmission on the devices connected to the link, and OC refers to the actual optical signal that is propagated over the fiber.
- OC-12: light sources that operate at a rate of 622 Mbps (STS-12).
- OC-48: light sources that operate at a rate of 2.5 Gbps (STS-48).

4.3 Information Technologies used in Campus Networks

Table 4 lists IT based on the following categories: tutorial, exploratory, application, and communication.

Classification of Education Technologies		
Category	**Definition**	**Examples**
Tutorial	Systems designed to teach by providing information, demonstrations, or simulations in a sequence determined by the system.	Computer-assisted instruction (CAI), Intelligent CAI, Instructional television, some videodisc/multimedia systems
Exploratory	Systems designed to facilitate student learning by providing information, demonstrations, or simulations when requested to do so by the student.	Microcomputer-based laboratories, microworlds/simulations, some videodisc/multimedia systems
Application	General-purpose tools for accomplishing tasks such as composition, data storage, or data analysis	Word processing software, spreadsheet software, database software, desktop publishing systems, video recording and editing equipment
Communication	Systems that allow groups of teachers and students to send information and data to each other through networks or other technologies	Local area networks, wide are networks, interactive distance learning

Table 4: IT Categories used in Education

Figure 18 shows the conceptual framework of the broad range of technological-management-related factors impinging on the development of network-based educational systems and the causal relationships among them.

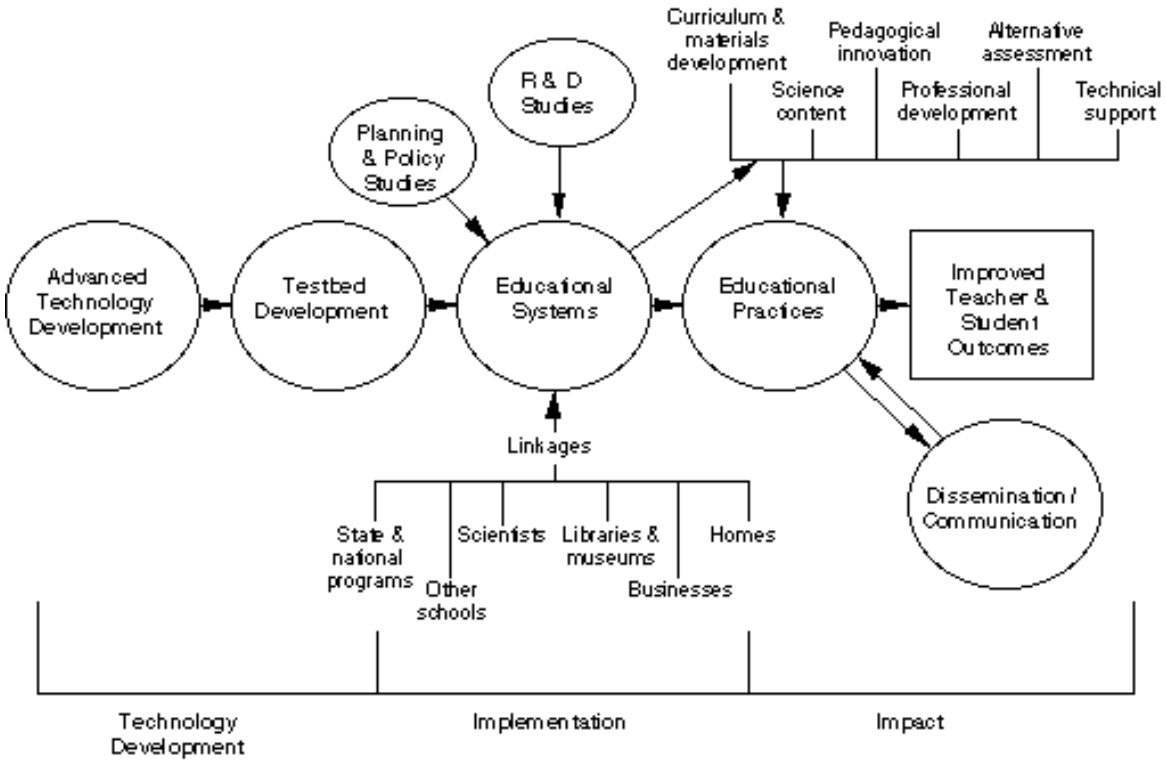


Figure 18: Causal Relationship between elements of an Educational Systems

Our conceptual framework suggests a minimalist approach to the idea of building smart machines that improves knowledge processing. Instead of attempting to build relatively complete knowledge representations, reasoning capabilities and/or pedagogical agent functionality, this alternative approach provides machines with minimal abilities (building blocks) to respond to the semantics of knowledge seeker activities and constructions, tests the educational value of these abilities, and adds functionality as needed to address deficiencies in the utility of the system.

4.4 Campus Networks and Internet2

A technology that is being offered to campus networks throughout the world is called Internet2 which began as a project of a relatively small group of research universities in late 1996, and has evolved into a sustaining membership organization of over 300 members and growing - universities and research centers are fast realizing that it is important to have fast access to knowledge [5].

Internet2 is developing and deploying advanced network applications and technologies for research and higher education institutions, thus accelerating the creation of tomorrow's Internet. It recreates the partnerships among academia, industry, and government that helped foster today's Internet in its infancy.

The primary goals of Internet2 are to:

- Create a leading edge network capability for the national research community
- Enable revolutionary Internet applications
- Ensure the rapid transfer of new network services and applications to the broader Internet community.

The investments by universities in Internet2, coupled with the efforts of industry and government, are helping to develop technologies such as IPv6, multicasting, and Quality of Service that will enable a new generation of Internet applications, benefiting all sectors of society. The wisdom of this approach to network research has been proven by the rise of today's Internet from the academic and federal research networks of the 1980s. Internet2 will replicate this success by relying on the university community for pre-commercial development.

4.4.1 Participating Universities

Internet2 participation is open to any university that commits to providing on-campus facilities that will allow advanced applications development. The financial investment required to meet these obligations may be more than many institutions can manage right now. Table 5 shows membership fees for the year 2004.

2004 Internet2 Membership Dues		
University	**Corporate**	**Affiliate**
$27,000	$27,000 (annual revenues > $1billion) $10,800 (annual revenues <$1billion)	$10,800

Table 5: 2004 Internet 2 Membership Fees

Internet2 intends to rapidly transfer the developments to the broader networking community. The cost of technology used and developed by Internet2 will drop within the reach of any institution that currently has a commodity Internet connection. Fifteen years ago, connecting to the Internet was frequently as expensive as participating in Internet2 is today. As the technology dropped in price, the efforts of the initial participants in network research spread out to reach the entire academic community. Deployment of Internet2 will follow a similar pattern.

Figure 19 shows a typical connection between two campus networks via Abilene.

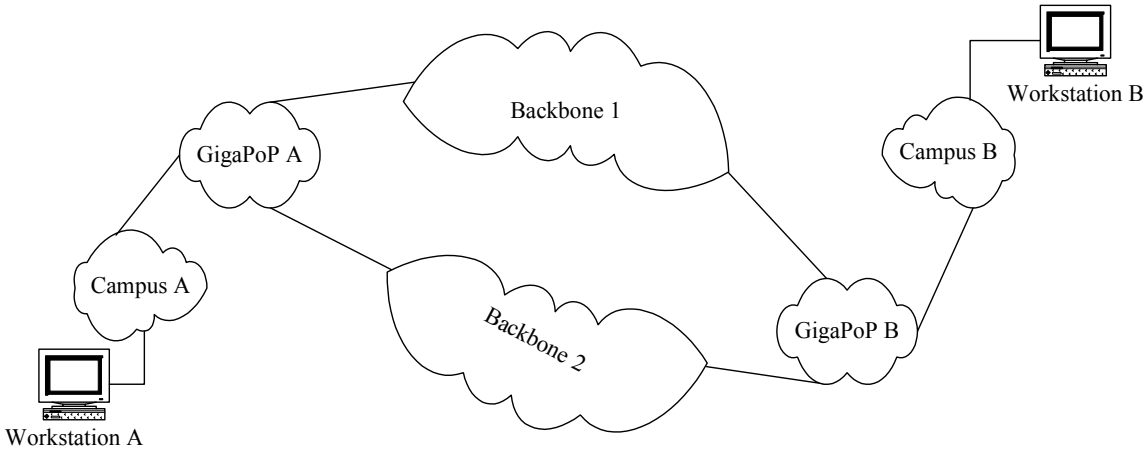


Figure 19: Two campus networks interconnected via backbones

A GigaPoP is an aggregate point of presence (POP) with gigabit capacity. While traditional Internet NAPs provide connections between networks at T1, T3, OC-1, OC-3, and occasionally OC-12 speeds, GigaPoPs are designed to provide access at much higher speeds so that different networks can exchange data at much higher rates of speed. GigaPoPs also usually provide a wider range of services than traditional NAPs (Network Access Points), which are primarily just layer-2 data exchange points - national ISPs (Internet Service Providers) connect together and exchange data at NAPs.

Traffic from various networks is aggregated at the GigaPoP and then routed to Abilene. This hierarchical model facilitated the shared use of distributed resources and minimizes the manual effort necessary to manage the network. Internet2 engineering committee recommends, among other factors, that packet loss within the GigaPoP is close to zero.

Figure 20 show the basic structure of a GigaPoP as a set of four high-speed switches (e.g., ATM switches), connected to two high-speed routers.

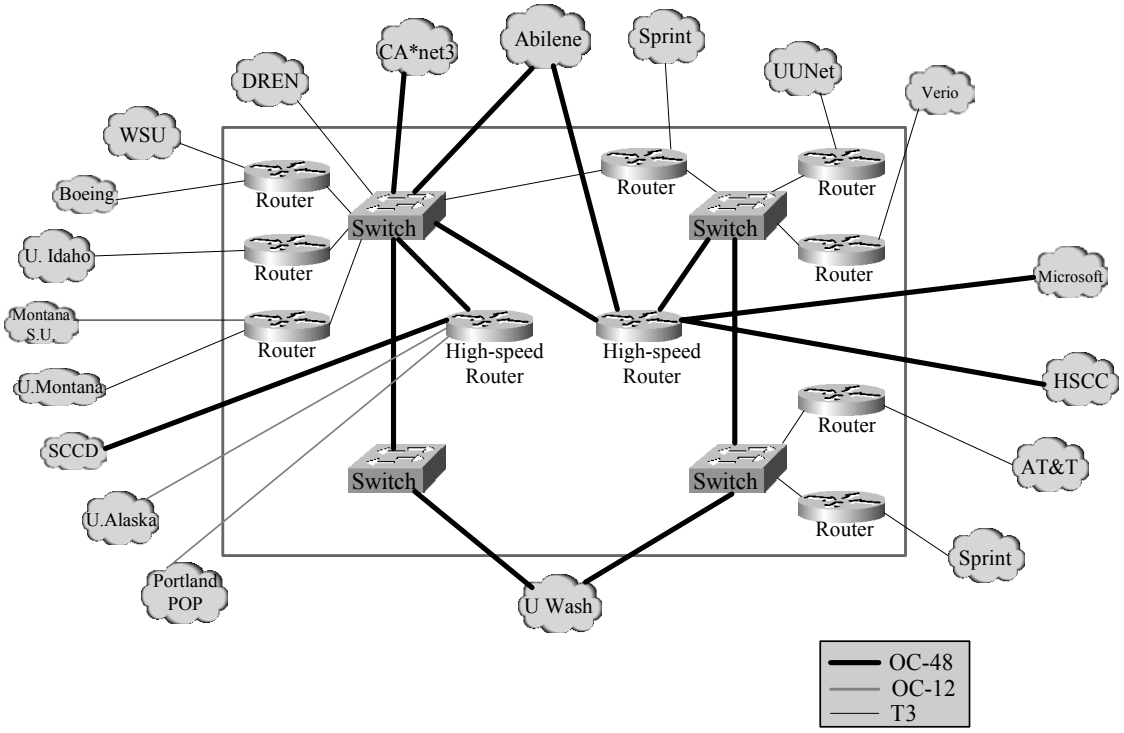


Figure 20: Inside the Pacific/Northwest GigaPoP. DREN = Defense Research Engineering; HSCC=High Speed Connectivity Consortium; OC=Optical Carrier

Abilene connects directly into this set of six core devices, while lower-speed networks connect into the core via a set of routers. Currently there are approximately 30 GigaPoPs connected to Abilene. These GigaPoPs service more than 70% of the participants. On average there will be 10-20 networks connected to a GigaPoP. The number depends on the interface provided by the ATM switch and router.

4.4.2 Key Technologies that make up Internet2

The main technologies that make up Internet2 are:

- *IPv6*: The successor to IPv4. Enhancements to IPv4 have addressed at least some of the problems of speed and service quality on the existing internet.
- *QBone*: The part of the Internet2 consortium looking at Quality of Service (QoS) issues. The DiffServ architecture, put forward by the Internet Engineering Task Force, is becoming acceptable as a cost-effective technique used to differentiate traffic classes in a network, thus allowing for quality of service for specific applications used in the network. We will return to Diffserv when we get into our model, which will include Diffserv as the means to have quality service in our network.
- *Digital Video Initiative (Internet2-DV)*: A project to build a high-quality digital video network, with the initial aim of delivering specialized academic research information.
- *Distributed Storage Initiative (DSI)*: A strategy based on server channels, or Internet Content Channels, which deliver replicated collections of files to end users. Files are replicated over multiple domains and multiple servers, but a single URL is used to access a given file, regardless of the actual domain of the server called to service the request.

4.4.3 Future Prospects for Internet2

Internet2 will not replace the Internet. Rather, its goal is to develop technology that will enhance the current Internet. Universities maintain, and continue to experience substantial growth in use of existing Internet connections, which they still obtain from commercial providers. The commercial sector is a full partner in this project, and will benefit from applications and technology developed by Internet2 members.

Just as the World Wide Web is the legacy of earlier investments in academic and federal research networks, the legacy of Internet2 will be technology adopted not only by universities, but also by the commercial sector.

A key goal of this effort is to accelerate the technology transfer necessary to move the appropriate technologies into the commercial sector-thus creating the basis of a next-generation network that will continue the United States' leadership in this important area.

Internet2 will also share its discoveries with others in the educational community-K-12, libraries, and other institutions. Internet2 is developing structures to share its experience and expertise with others in the education community and beyond. This is the approach that characterized the first Internet and it can work again today.

4.4.4 ABILENE

The Abilene Network is an Internet2 high-performance backbone network that enables the development of advanced Internet applications and the deployment of leading-edge network services to Internet2 universities and research labs across the country. The Abilene Project begun in 1998 and is led by the University Corporation for Advanced Internet Development (UCAID), working in collaboration with Qwest Communications, Cisco Systems, Nortel Networks and Indiana University.

Abilene initially started operating at speeds up to 2.4 gigabits per second (Gbps) and in 2004 it was upgraded to 10 Gbps. The new OC-192 circuits provide Abilene participants and its peer networks with leading edge networking capability. Figure 21 shows all the main nodes in the domestic Abilene network.

Figure 21: Abilene domestic peers

The Abilene Network supports the development of applications such as virtual laboratories, digital libraries, distance education and tele-immersion, as well as the advanced networking capabilities that are the focus of Internet2. Abilene complements and peers with other high-performance research networks in the United States and internationally.

Abilene networks provide the academia with the opportunity to link their operations through giga-stream channels, as if they were in the same geographical place, and at the same time ensure 24-hour servicing to any user.

The two interconnected campus networks in this research, national and global, consist of one that spans the USA, and the other one that connects this network with some nodes across Europe, Asia, and South America. The idea behind is to quantify the difference in performance of these two campus networks with the same type of traffic, different connectivities, varying network parameters, and use the statistics generated by the network simulation to evaluate their overall performance under different scenarios. Each design will be simulated and the results of the simulation will be validated against real values taken from the Abilene network website.

Figure 22 shows the map of the Abilene's international network peers. Peer networks are domestic or international research and educations networks that peer with the Abilene Network.

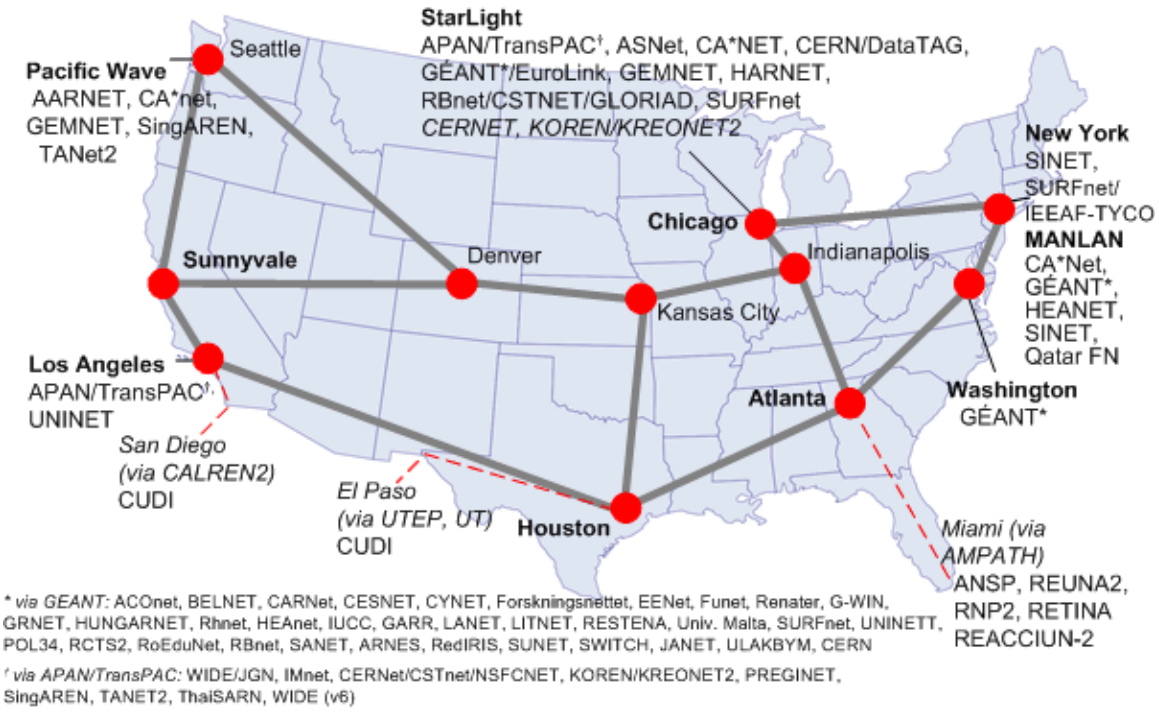


Figure 22: Abilene International Network Peers

Through mutual agreements with similar organizations around the world, Internet2 establishes ties that help to ensure the continued global interoperability of advanced networking, and enable collaboration between U.S. researchers, faculty, students and their overseas counterparts.

These U.S. and international peer networks peer with the Abilene network at various exchange points in U.S. Peer networks are dedicated to high-performance research and education.

These peer organizations engage in projects similar to those of Internet2 in scope and objective, and most also serve similar constituencies – universities and other non-profit institutions committed to advancing networking technology and applications for use in

the education and research community. The goal of these international relationships is two-fold: to ensure global interoperability in the next generation of networking technologies, and to enable research and education through advanced applications on a worldwide scale. A high-speed, high-capacity "network of networks" with global reach will promote unprecedented collaboration between U.S. researchers, faculty, students, and their overseas counterparts.

A list of some of the Abilene's international peer networks is shown in table 6 with their respective bandwidth. The actual number for some of the nodes in our networks is taken from some of these networks: ANSP (Brasil), GEANT (Europe), TRANSPAC (Japan).

Peer Network	MoU Partner	Country/Region	Interconnect	Bandwidth (Mbps)
ANSP	RNP2	Sao Paulo, Brasil	Atlanta	45
CA*net	Canarie	Canada	Seattle	3,000
CERN	N/A	Europe	Chicago	10,000
CERNET	CERNET	China	Seattle	622
GEANT	DANTE	Pan-European	New York	7,500
TRANSPAC	APAN	Japan	Los Angeles, California	155

Table 6: Abilene's International Peer Networks

5. Simulation

A simulation is a representation of certain features of the behavior of a physical system or abstract model of a real system [13]. Simulators are systems comprising of either software and/or hardware, and are developed to apply the simulation process. There are other available techniques that can be used in systems planning such as simple calculations, spreadsheets, operations research techniques (such as linear programming and queuing theory), and special computerized tools for scheduling, layout, and so forth. While these tools can provide quick and approximate solutions they tend to make oversimplifying assumptions, perform only static calculations, and are limited to narrow classes of problems [19]. Additionally, they fail to fully account for interdependencies and variability of complex systems and therefore are not as accurate as simulation in predicting complex system performance, as shown in figure 28.

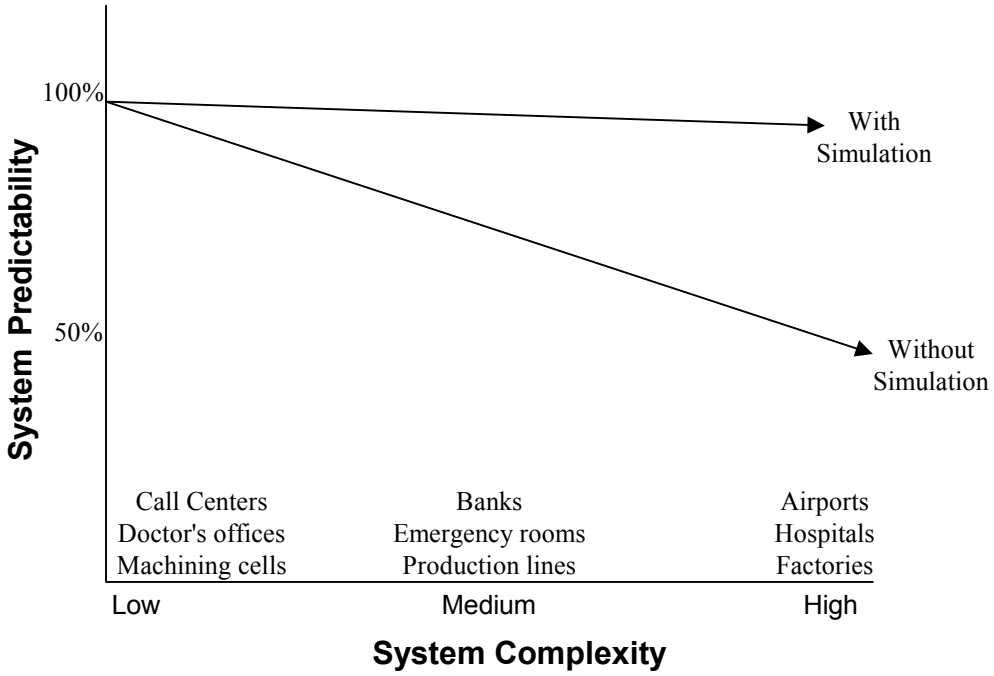


Figure 23: System Predictability vs System Complexity

5.1 Modes of Simulation

Simulation has two basic forms: *discrete* and *continuous*. In discrete simulation, the changes in a system over time are represented as a series of instantaneous occurrences of *events.* These are reflected by changes in the *state variables* that describe the current *system state* [17].

Discrete event simulations typically model dynamic processes that occur over particular time duration with events representing state changes that occur at specific points in time. A simulation clock is used to keep a record of the simulated time, and events are generally time-stamped with the simulation time at which that event is scheduled to occur. Discrete simulation enables a more detailed range of data to be collected than continuous simulation. For example, consider a queue in a bank. In a continuous simulation the movement of customers through the queue would be modeled as a flow or rate, so only parameters such as mean arrival rate would be applicable. However, in a discrete event simulation, the movement of each individual person would be modeled and hence (rare events' such as the refusal to admit individual customers could be studied.

5.2 Time Advancement and the Event List

Some way of storing all the foreseeable future scheduled events is required. This is accomplished by the *event list* which normally takes the form of a time ordered sequence of event records with the next scheduled event at the head. The event list can be sited within the logical architecture of the simulator according to one of two schemes: *centralized event list* and *distributed event list*.

In centralized event list architecture, one large event list is used to store all of the events for the whole simulation, whereas with a distributed event list there is a list for each model in the simulation that stores all of the locally scheduled events.

There are two principle schemes for advancing the simulation:

- time driven simulation, and
- event *driven simulation*.

In both of these schemes the simulation essentially progresses through a process of the next event being taken from the event list, processed, and any resultant events being put back in their correct time position in the list. Where they differ is in the way that the simulation clock is advanced.

In time driven simulation fixed-increment time progression is used. Here, the simulation clock advances by constant amounts, or *ticks*. After each tick, all of the events that were scheduled to occur during that time interval are processed, and the system state updated.

This scheme is generally used if events are known to occur at fixed regular intervals but it can be inefficient if the tick is set incorrectly. If the tick is set to small there will be only a low probability that an event will be scheduled for a given tick. Therefore, there will be many ticks in which no useful processing is done in the simulator. However, if the tick value is set too large, then the error resulting from the fact that all events scheduled for that tick are processed at the end of the tick will be unacceptably high.In event driven

simulation a next-event time progression scheme is used. This is the most common technique used in network simulation because it can cope with random event times. In this scheme, after the processing of an event, the simulation clock is advanced to the time of the next scheduled event in the event list. Therefore, periods of inactivity are skipped and processor time is not wasted waiting for future events to occur.

The sequence of operations that occurs in an event driven simulator is as follows:

- During system initialization, the simulator configuration is loaded, simulation parameters and measurements initialized, the simulation clock reset, and each of the components of the system polled to determine when they expect to generate their first event,
- Based on this the event list is initialized,
- The simulation then enters a loop in which the earliest event in the event list is processed, the system state updated, and any resulting events placed back in the event list in correct time order,
- A check is then made to see if some stop criterion has been met which could be some end simulation time, or it could be based on the state of some other system variable. If this criterion has not been met, then the next event in the event list is fetched and processed,
- Once the stop criterion is satisfied, the loop exits and the final results of the simulation collected.

Figure 29 illustrates, as per the ITU-T Specification and Description Language (SDL), the sequence of operations that occurs in an event driven simulator (Z.100 only applies to linear processing of events, as opposed to parallel processing of events by a simulator):

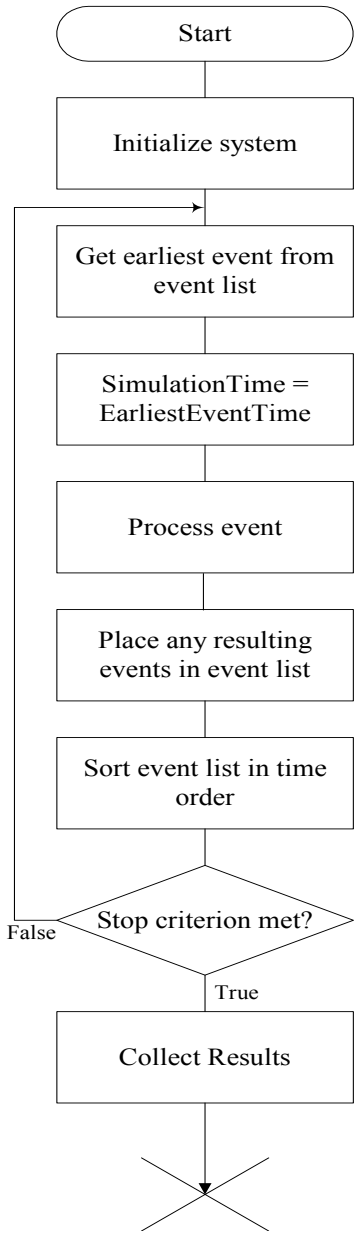


Figure 24: Sequence of Operations in an Event Driven Simulator

5.3 Event Generation and Random Number Generators

Event times can be generated in one of two ways. Measurements from a real system can be used to determine the times of the events. However, whilst this method can offer authenticity, it has the disadvantage that for reasonable length simulations a great deal of data must be stored. Furthermore, it implies that real systems already exist to provide such data to experimenters. Clearly, this is a problem in relation to campus simulation since the real systems do not necessarily already exist, and even if they do access to data from them is often difficult, costly and time consuming.

An alternative method of event generation is to use random number generators coupled with models of the sources (e.g. probability distributions of packets inter-arrival times). From these, particular event times can be calculated. However, it is important to note that random number generators are usually pseudo-random and so care must be taken to ensure that these times are adequately realistic. A good pseudo random number generator should produce a sequence of numbers, evenly distributed, which should not exhibit any correlation between each other. The sequence should also be reproducible, which aids in debugging and can be used to increase the precision of results [18] and should be as long as possible. After the random number generator sequences through all random numbers in its cycle, the cycle starts over again through the same sequence. The length of the cycle before it repeats is called the cycle period, which is usually extremely long (the cycle length for the random number generator used in OPNET is over 2.1 billion).
A review of random number generator algorithms is given by Law & Kelton [15], an in depth discussion of these being beyond the scope of this thesis.

5.4 Transient and Steady States

When a simulation is first started any measurements made will be partially dependent on the initial states of the system variables. After some period, the simulation will settle down to a steady state at which the distribution of measurements (but not necessarily the actual measurements themselves) will become stable. This initial state is known as a transient state. It is not just applicable to the period immediately following the start of a simulation run, but also following major occurrences when, for example, new sources start generating network traffic. The precise nature of that being measured using the simulator will determine whether the steady or the transient state is of interest. In general, the study of data packet transmission such as packet losses or packet delay will require that the simulator has reached the steady state in order to be certain of statistically significant results. However, if communication link conditions, such as link delays are of interest then clearly the transient state will be more relevant. In either case, it is often difficult to judge the exact point at which the transient state ends and the steady state begins.

The non-terminating aspect of an actual campus network daily operation can still be analyzed by a finite amount of either simulation time or a relatively small amount of actual traffic. This means that practical use of network simulation requires some economies of scale. Also, our campus networks operations begin with a warm-up (or transient) state and gradually move to a steady state. Once the initial transient phase has diminished to the point where the impact of the initial condition on the system's response is negligible, we consider to have reached steady state. Statistics collected during the transient period (first 110 seconds) are discarded to avoid errors in the results.

5.5 Simulation Languages and Tools

The choice of the language in which the simulator is written can have a significant impact on both its performance and on the development time. Some simulators are written in general purpose programming languages. For example, MICROSIM and LINKSIM are written in PASCAL while the ICM simulator, MADS, is written in ANSI C. Despite the widespread use of conventional programming languages, several specialized languages and tools exist that can be used to generate the simulation model. One example is the SIMULA language in which real systems are modeled as a number of interacting processes. Object oriented languages, such as C++, have also been used to write simulators. A number of specialized commercial simulation tools have emerged that allow the user to define, the network using a graphical interface. For example, the OPNET' [13] development environment allows the user to define the network in terms of nodes, process models and state diagrams. This description is then used to generate the appropriate C code, which is finally compiled to produce a self contained simulator. Other examples of such commercial packages include MODSIM, NST2 and BONES. Whilst such approaches can give rise to considerable savings in the development time for a simulator, as in the case of specialized simulation languages, the code generated is often inefficient and hence run times can be relatively long when compared with hand-crafted simulators written in a conventional programming language.

Due to our limitation of computing resources, this thesis is also concerned with methods for reducing the run time of campus network simulation and not development time. Our simulator, OPNET, suggests using the optimized kernel, which in fact reduce quite considerably (few hours) the time it took to simulate our 10 minutes runs.

5.6 OPNET

OPNET stands for Optimized Network Engineering Tools. It is a workstation-based application for the modeling and simulation of communication systems. OPNET allows the user to specify the system in terms of incrementally decreasing levels of abstraction from the sub-net level down to the individual process level. This is done using a graphical user interface (GUI). These specifications are then compiled to produce a simulator executable that can be run, either under the direct control of the OPNET GUI or independently. Debugging and measurement functions can be embedded within the simulator executable [14].

The network simulation environment in OPNET is organized in a hierarchical manner consisting of three structural levels: the *network model* level, the *node model* level, and the *process model* level.

At a given level in the hierarchy, a model is configured or defined in terms of the models at the next lower layer. This means that a network model will consist of a group of node models arranged and connected in a certain way.

Likewise, a node model consists of one or more process models connected in a certain manner. The node models typically represent the protocol stack underlying the network devices.

The process models in turn are finite-state machines (FSMs) built using C/C++ and Proto-C (which uses OPNET-defined simulation kernels or functions). Thus, each model can be imagined to be an object, each with a set of attributes that can be modified.

Hence, OPNET can be described as being object-oriented in its approach to modeling components within simulations.

The OPNET Modeler™ includes an extensive library of built-in models and functions, many of which have been utilized in building the simulation models for this study. This library includes models of the most widely used communications protocols such as: Ethernet, IP, UDP, TCP, ATM, FDDI, Frame Relay and CDPD. These models are very detailed and include all features of the mentioned protocols. OPNET also contains models of the most commonly used routers and switches (for example a number of Cisco routers, which we are going to make use of in our network model).

When developing an OPNET model the user can either:

- Use a ready-made model, if one exists a ready-made model for the protocol or network that he wants to model.
- Use ready-made models such as Ethernet, IP, and TCP models as a part of his model as parts of an FTP model.
- Develop a completely new model from scratch.

In the workspace, you can create a network model, collect statistics directly from each network object or from the network as a whole, execute simulations, and view results. OPNET also allows the user to collect data or results by selecting appropriate statistics before each simulation run.

5.6.1 Modeling using OPNET

An overview of the simulator development process using OPNET is given in Figure 30.

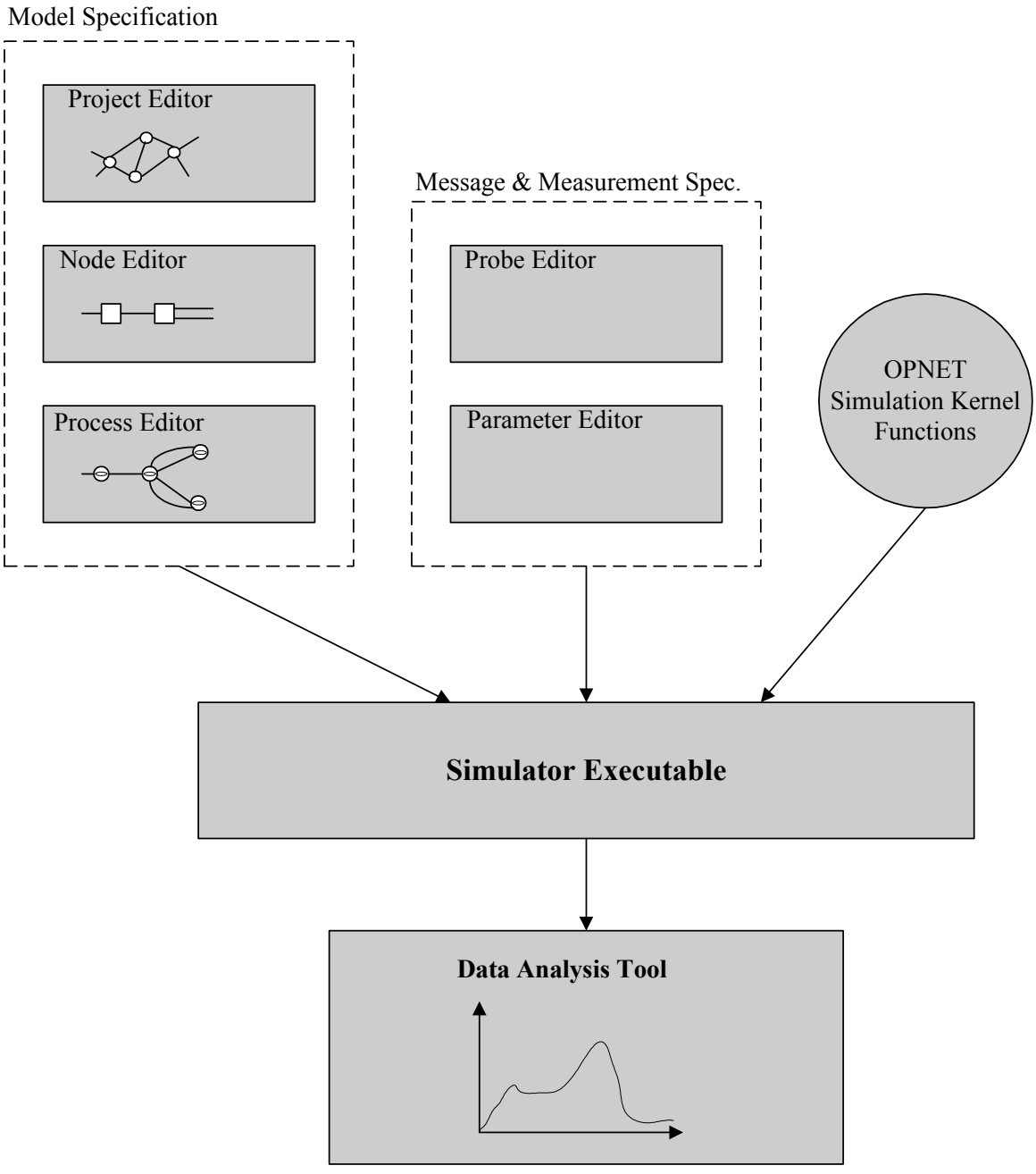


Figure 25: Simulation Development Process with OPNET

The user specifies the network to be modeled using a top-down approach. For the basic aspects of modeling, several editors are needed to accomplish a simulation project. Basically, they are:

Project Editor:

The Project Editor is the main staging area for creating a network simulation.
From this editor, user can create a network model using models from the standard library, collect statistics about the network, run the simulation, and view the results. However, there are some functions cannot be done in the Project Editor, such as creating node and process models, building packet formats, and creating filters and parameters.

Node Editor:

The Node Editor is used to define the behavior of each network object. Behavior is defined using different modules, each of which models some internal aspect of node behavior such as data creation, data storage, etc. Modules are connected via packet streams or statistic wires.

Process Model Editor:

The Process Editor is used to create process models, which control the underlying functionality of the node models created in the Node Editor. Process models are represented by finite state machines (FSMs), and are created with icons that represent states and lines that represent transitions between states. Operations performed in each state or for a transition are described in embedded C or C++ code blocks.

Probe Editor:

The Probe Editor is used to specify the statistics to be collected during simulation. While this can be done in the Project Editor, the Probe editor can be used to set additional characteristics of each probe. There are several different types of statistics that can be collected using different probes, including global statistics, link statistics, node statistics, attributes statistics, and several types of animation statistics.

Analysis Tool:

Although simulation results can be viewed in the Project Editor, the Analysis Tool has several useful additional features. You can, for example, create scalar graphs and parametric studies, define templates to which you apply statistical data, and create analysis configurations for application of statistical data, and create analysis configurations that can be saved and viewed later

The user can also specify the format of intra-model and inter-model messages (packets or interrupts) using the Parameter Editor and also the measurement statistics that are recorded in output scalar or output vector files for later analysis. Once the network model has been fully defined, it is compiled and linked with the OPNET simulation kernel to produce a simulation executable. This executable can be either run as a stand-alone simulation or run under the control of OPNET. Output vector and output scalar files generated during the simulation run can be analyzed using the Analysis Editor.

5.6.2 Model Validation using OPNET

Once a simulation has reached the stage where it can be successfully compiled and executed, it is still too early to start gathering statistics and making interpretation on this data. While the simulation is syntactically correct (i.e. it compiled), it may not be semantically correct. That is, the simulation may not accurately simulate the behavior of the actual system being modeled. There are essentially three techniques that an OPNET user can use to perform model validation: (we did not use the first one)

The OPNET debugger (ODB)

The ODB allows a user to interactively control the progression of a simulation. The user can define breakpoints and specify the nature of debugging information to be generated. This tool can be used to track the exact time and nature of an error, which may or may not result in the simulation halting.

Automatic Animation Generation

Animations are useful to graphically visualize a model's behavior. Animations can be run synchronously (i.e. as a simulation executes) or asynchronously (i.e. after simulation execution). A simulation can be debugged in parallel with a synchronous animation.

Output Data Generation

Another approach to model validation is to generate statistical output data. While this information might not be the ultimate statistics of interest, it can prove useful in identifying problems with a simulation or simply understanding a simulation's behavior.

5.6.3 Network Characteristics

Attribute selection has to be done for each and every element of the network. In OPNET we use the attribute menu selector option for each element. First we need to define the application and profile definition of the network.

5.6.3.1 Application Definition Object

A user's behavior or "profile" can be described by the applications he or she uses, for how long, and how often these applications are used each day. A number of common campus network applications such as Email for sending/receiving electronic messages, FTP for sending and receiving files, HTTP for Web browsing, and database queries have been selected for this project. While multimedia application is present in campus universities their use is being restricted by the campus network administrator due to the fact they consume most of the available bandwidth and also there is a factor of legality of what they are used for among campus users.

5.6.3.2 Profile Definition Object

Profile definition relates each application with its respective clients. The type of profile used in this project is denominated a "Campus Client". A campus client is an Ethernet client profile consisting of four "heavy-loaded" applications: email, FTP, Web browsing, and database query. The reason for the "heavy-loaded" characteristics of our campus applications is a matter of scaling our model so the load would be of approximately equal magnitude to the existing load of actual campus networks, such as, those of some of the Abilene network peers.

6. Queuing Theory

Many systems of interest in a simulation study contain processes in which there is a demand for service that causes congestion. The system can service entities at a rate which, in general, is greater than the rate at which entities arrive, but there are random fluctuations either in the rate of arrival, the rate of service, or both. As a result, there are times when more entities arrive than can be served at one time, and some entities must wait for service. The entities are then said to join a waiting line or, more simply, to form a queue.

Queuing theory is a static analysis technique that provides good estimates for basic problems such as determining the expected average number of entities in a queue or the average waiting times for entities in a queuing system. In plain terms, queuing theory is essentially the science of waiting lines, and it is also essential in analyzes of computer systems and computer networks. This theory is used with Priority Queuing (delay), Time-Sharing systems, Multi-Access systems, Analysis and Design of Communication Networks (traffic flow, packet switching, etc), Measurement and Flow Control (routing). Figure 31 shows a queuing system: one or more queues and one or more servers.

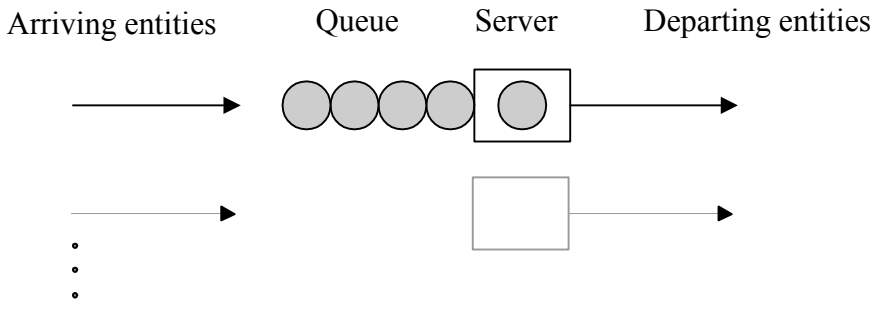


Figure 26: Queuing System

Entities, referred to in queuing theory, enter the queuing system and either are immediately served if a server is available or wait in a queue until a server becomes available. Entities may be serviced using one of several queuing disciplines: First-in, first-out (FIFO); last-in, first-out (LIFO); priority; and others. Most network devices use a "first-in, first-out" (FIFO) rule to keep traffic moving. However, there are two software methods to change the rules so that traffic is "first out" because of its higher priority, not because it was "first in":

. Set network devices to use one of the queuing schemas.

. Use a quality of service (QoS) mechanism that guarantees which data traffic receives priority over other data traffic.

The entire network does not need to guarantee the bandwidth; rather, the devices in the path between the end-station running an application and the application server must be able to guarantee traffic priority. In most cases, a router is the device to be set up with traffic guarantees, since switches do not understand or care about traffic classification.

To model the queuing process of a system, the probability functions that describe the arrival patterns and service times must be given. We assume an M/M/1 process for our campus networks, which means: the interarrival and service times are distributed exponentially, and the arrivals come to the system according to a Poisson distribution. The core routers are configured as priority-enable devices in one of the three scenarios, while for the other two scenarios the routers use FIFO as the queuing schema. We then analyze traffic under these different scenarios – expecting to see improved response times for the applications with higher priority.

6.1 Measures of Queues

Two principal measures of queues are the mean number of entities waiting and the mean time they spend waiting. Both these quantities may refer to the total number of entities in the system, those waiting and those being served, and they may refer to the entities in the queue only. To determine the expected number of entities, the expected system time and expected waiting time, it is necessary to gather some knowledge about the probability density function (PDF) of the service time (amount of time it takes for an entity to be out of the queue and leave the system) or interarrival time (the time which passes between two arrivals in the queue). The system capacity, or number of entities allowed in the system at any one time, may be either finite or, as is often the case, infinite. Different interarrival time distributions (such as constant or exponential) may also be analyzed, coming from either a finite or infinite population (entities). Service times may also follow one of several distributions such as exponential or constant.

Kendall [21] devised a simple system for classifying queuing systems in the form A/B/s, where A is the type of interarrival distribution, B is the type of service time distribution, and s is the number of servers. Typical distribution types for A and B are:

M - for Markovian or exponential distribution

G - for a general distribution

D - for a deterministic or constant value

An M/D/1 queuing system, for example, is a system in which interarrival times are exponentially distributed, service times are constant, and there is a single server.

The arrival rate in a queuing system is represented by lambda (λ), the service rate by mu (μ), the mean interarrival time then becomes $1/\lambda$ and the mean service time is $1/\mu$. A traffic intensity factor λ/μ is a parameter used in many of the queuing equations and is represented by the rho (ρ). Common measures in a queuing system are based on steady-state or long-term expected values and include:

L = expected number of entities in the system (number in the queue and in service)

L_q = expected number of entities in the queue (queue length)

W = expected time in the system (flow time)

W_q = expected time in the queue (waiting time)

P_n = probability of exactly n customers in the system (n=0, 1, ….)

The M/M/1 system with infinite capacity, single server queue with a Poisson arrival process, exponential distribution time for service time, and a FIFO queue discipline is perhaps the most basic queuing problem and sufficiently conveys the procedure for analyzing queuing systems and understanding how the analysis is performed.

The equations for calculating the common performance measures in an M/M/1 are:

$$L = \frac{\rho}{(1-\rho)} = \frac{\lambda}{(\mu-\lambda)}$$

$$L_q = L - \rho = \frac{\rho^2}{(1-\rho)}$$

$$W = \frac{1}{\mu-\lambda}$$

$$W_q = \frac{\lambda}{\mu(\mu-\lambda)}$$

$$P_n = (1-\rho)\rho^n \qquad n = 0, 1, \ldots..$$

If either the expected number of entities in the system or the expected waiting time is known, the other can be calculated easily using Little's Law [22]:

$$L = \lambda W$$

Little's Law also can be applied to the queue length and waiting time:

$$L_q = \lambda W_q$$

Little's Law says that the average number of entities in a queuing system is equal to the average arrival rate of entities to that system, times the average time spent in that system. Queues can be used to represent buffers, for example in switches, while servers represent the routing elements (table & ports) of switches.

6.2 Poisson Processes

Queuing processes where the arrivals are said to be completely random, have the characteristic that an arrival can occur at any time, subject only to the restriction that the mean arrival rate be some given value. More formally, it is assumed that the time of the next arrival is independent of the previous arrival, and that the probability of an arrival in an interval Δt is proportional to Δt.

Processes that have these properties can be represented by a Poisson distribution.

Lots of physical phenomena are described by this process, such as radioactive particles decay, telephone calls received at a switchboard during a fixed period of time, molecular distributions. stellar distributions, geographical distribution of plants, etc. Again, these are processes that occur randomly, their interarrival times as well as their duration are exponentially distributed, but their rate of occurrence is constant.

Simply stated, Poisson distribution is a method of determining the probability of occurrence of an unlikely event within a large number of independent trials. For example, if λ, the average number of users at any time on a network is known, Poisson distribution can be used to determine the minimum or maximum number of users in a certain time period.

The reason for the success of the Poisson model is the analytical solvability, whereas the other models often need numerical calculations and approximations.
The model is characterized by only one parameter lambda (λ), the mean number of events per second. This parameter can easily be determined, for example, the overall mean packet rate can be calculated of a packet trace, using TCPDUMP program, of LAN traffic on a small subnet.

An example of Poisson distribution, figure 27, shows a discrete distribution bounded at 0 on the low side and unbounded on the high side with a mean rate of 6.7.

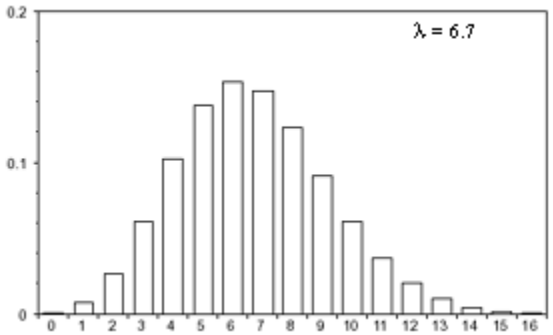


Figure 27: Poisson distribution (6.7)

The mathematical formula of Poisson distribution is:

$$P(x) = \frac{e^{-\lambda}\lambda^{x}}{x!} \quad ,$$

where x = number of occurrences and λ = mean rate of occurrence.

Note that the time between arrivals is exponentially distributed. In particular, although the Poisson distribution has been described as a representation of a random arrival pattern, it is also used to describe a service pattern. When a service time is considered to be completely random, it may be represented by an exponential distribution. In that case, the associated Poisson distribution represents the number of entities served within a given interval. Poisson distribution can be used to represent packet arrivals at a network node. The arriving packets may come from many different nodes, therefore are independent of each other. Hence the number of arrivals within a time interval is independent of other arrivals, depending only on the length of the time arrival. Let λ be the average packet arrival rate and μ the maximum departure rate. We define load ρ to be given by $\rho = \lambda/\mu$. Clearly, we want the arrival λ rate to be less than the departure rate μ, and hence $\rho < 1$.

The service time of a message depends on its size. If the length of a message has Poisson distribution, then the time it takes to service the packet has exponential distribution. Consequently, when studying the behavior of telecommunications networks, M/M/1 queues are utilized, implying that these queues have exponential interarrival and service times with a single server. The queue size is unbounded, with an infinite number of messages in the system and the service discipline is FIFO.

6.3 Queuing Schemas

Different scheduling algorithms can be used in networks. What they have in common is that they are memoryless – the queuing process does not retain the history status of the event just queued. These schemas include the following: FIFO and priority queues.

6.3.1 First-In-First-Out (FIFO)

The simplest queuing scheme used for best effort service. Packets are served in exact order in which they arrive regardless of their priority classes. Thus, it does not discriminate among traffic classes and cannot be used for QoS implementation. FIFO queuing is fast and easy but can make delay-sensitive applications wait behind long bursts of delay-insensitive data.

6.3.2 Priority Queuing (PQ)

With this scheme, the buffer space is prioritized. Packets are queued or placed into the buffer space according to the packet priority. High-priority packets are sent on the output link before lower-priority packets. The problem with this scheme is that it is possible that lower-priority packets are delayed too long to be useful, or possibly, they are discarded.

6.3.3 Weighted Fair Queuing (WFQ)

WFQ with random early discard (RED) is a technique that attempts to intelligently manage buffer space by influencing the type of traffic (priority) that is allowed into the network. WFQ provides bandwidth allocations and delay bounds to specified traffic types. It does not require users to configure router buffer allocations. Rather, it uses a

servicing algorithm that attempts to provide predictable delays. As packets arrive, the buffer space is sorted in a way that attempts to ensure that each packet is delayed the appropriate amount of time. The RED mechanism is a scheme in that each router is programmed to monitor its own queue length and when it detects that congestion is imminent, to notify the source to adjust its congestion window.

Basically, WFQ forwards packets just before the packet reaches the maximum delay value for its priority. This approach can significantly improve response times and decrease the delay variation for interactive traffic, resulting in more predictable round-trip delays. Obviously, as more and more packets arrive, the task of timely packet delivery becomes more complex, and eventually, as the router becomes too congested to provide acceptable delays, RED begins dropping packets. RED selectively drops single packets from multiple flows, forcing the TCP sessions to go into slow-start mode, and reducing network traffic at its source.

6.3.4 Class-Based Queuing (CBQ)

CBQ is an enhancement to PQ. CBQ attempts to provide fairness by allocating portions of link bandwidth to specific types of traffic. Instead of providing absolute service to the high-priority queue, it deploys a more equitable scheme. With CBQ, the user configures the preference with which each queue will be serviced and the amount of traffic that will be delivered from the queue to the link on each pass. This method permits the user to assign buffer space to defined traffic classes, thus reserving link bandwidth. Typically, the buffer allocation is static, and the user must configure sufficient buffers in each router along each potential path to accommodate each of the traffic classes.

7. Quality of Service (QoS)

The explosive growth in the Internet over the last decade, along with simultaneous proliferation of new *bandwidth-hungry* applications has been putting compelling demands on today's networking world. This has led to the development of what is today known as the *quality of service (QoS)* technologies.

QoS is similar to queuing in that it designates bandwidth requirements (service availability) for high-priority traffic. But QoS goes beyond queuing in that it reflects the performance requirements (transmission quality) of such traffic. QoS works in conjunction with one of the queuing mechanisms described previously.

Quality of service can also be described as a set of technologies that enables network administrators or service providers to manage the effects of congestion on application traffic by using network resources optimally, rather than continually adding capacity.
It is precisely this characteristic of QoS that this study is employing to inject QoS into our campus networks in order to improve performance instead of provisioning the network.
From a user's perspective, QoS could be interpreted as the degree of satisfaction he derives from the service he is receiving. Alternately, QoS could be defined in terms of specific performance metrics such as – end-to-end delay, delay variation or jitter, packet loss ratio, throughput etc.
Our campus networks, which will not include multimedia applications, use Best effort models of QoS provision for all of the applications (HTTP, FTP, E-Mail) except for database queries. Best effort model is predominant in networks where multimedia service protocols are either disabled or rarely used.

The use of multimedia file-sharing and other bandwidth-intensive applications have the potential to consume large amounts of bandwidth [30]. While high-speed LAN campus is often capable of handling this traffic, campus connections to the Internet are particularly prone to congestion from the use of these applications. To deal with this problem, campus network managers usually suspend or limit file sharing and multimedia services during regular hours either for sake of bandwidth economy or for safety purposes. Fortunately, present analyses of campus data traffic, e.g. figure 33, reveals that multimedia traffic accounts for less than two percent of total traffic.

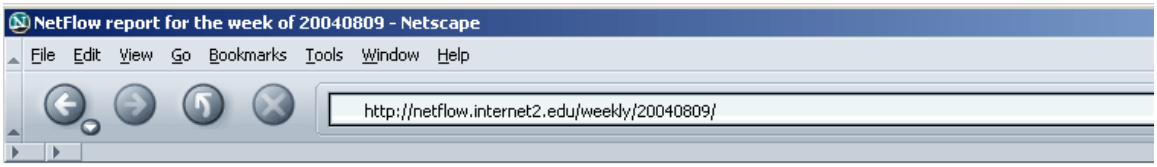


Table 6. Aggregated Application Types (Full Data Set)

Type	Octets		Packets	
Measurement	45.63%	196.9T	10.88%	39.05G
Data Transfers	19.25%	83.06T	26.15%	93.84G
File Sharing	5.91%	25.49T	10.67%	38.28G
Advanced Apps	4.11%	17.74T	5.99%	21.49G
Encrypted Traffic	3.37%	14.56T	4.83%	17.34G
Audio/Video	1.52%	6.551T	2.32%	8.336G
Misc	1.25%	5.389T	4.01%	14.38G
Games	0.39%	1.670T	0.83%	2.988G
Unidentified	18.58%	80.19T	34.32%	123.1G
Total	100.00%	431.5T	100.00%	358.8G

Figure 28: Abilene Weekly Traffic by Application Type

Nonetheless, multimedia traffic is posed to increase once it is given the appropriate conditions to do so, such as enough bandwidth and quality of service.

The tolerance of the quality of service parameters varies from application to application. For example, a file transfer is much more concerned with reliability than with delay variation, but a video application is much more concerned with delay and delay variation than with reliability.

Table 7 characterizes the Quality of Service requirements for some common applications.

	Delay	**Jitter (Delay Variation)**	**Reliability**
Mail	Low	Low	High
File Transfer	Low	Low	High
Database	Medium	Low	High
Multimedia (video, telephony, video conference)	High	High	Low

Table 7: Quality of Service Requirements

The current best effort approach to quality of service in IP networks can no longer satisfy a diverse variety of customer service requirements, and that is why there is a need for alternative strategies.

We believe that the IP networks needs means for providing a QoS that does not cause network congestion and keeps overall link utilization high.

In this paper, we emphasize that one of the best ways to achieve this goal is to introduce a differentiated service at the network boundaries.

7.1 QoS in IP Networks

QoS is a concept that is usually associated with connection-oriented networks like ATM and frame relay. However, it is becoming increasingly popular in IP networks.
IP networks are basically heterogeneous networks where each network segment can employ a different physical network media and a different data-link layer protocol. Such networks can only provide one simple service class to all uses with respect to quality of service (QoS): best-effort datagram delivery, which cannot provide any service quality guarantees. It works well with asynchronous applications such as file transfer and electronic mail where the applications have little requirement on QoS. However, newer applications, such as distributed multimedia applications, require specific QoS guarantees. Presently, data generated by such applications are carried by special services, such as dedicated point-to-point connections and/or virtual private networks, where arrangements with service providers allow for QoS at a price, obviously.

The IP packet structure provides for a differentiated service through one of its header fields: Type of Service (ToS), which is an 8-bit header field found in IP packets. Its main purpose is to specify reliability, precedence, delay, and throughput. ToS is supposed to indicate something special about the packet that is desired by the source and can intelligently be supplied by the routers. The IPv4 packet with a detailed ToS field is shown in figure 29.

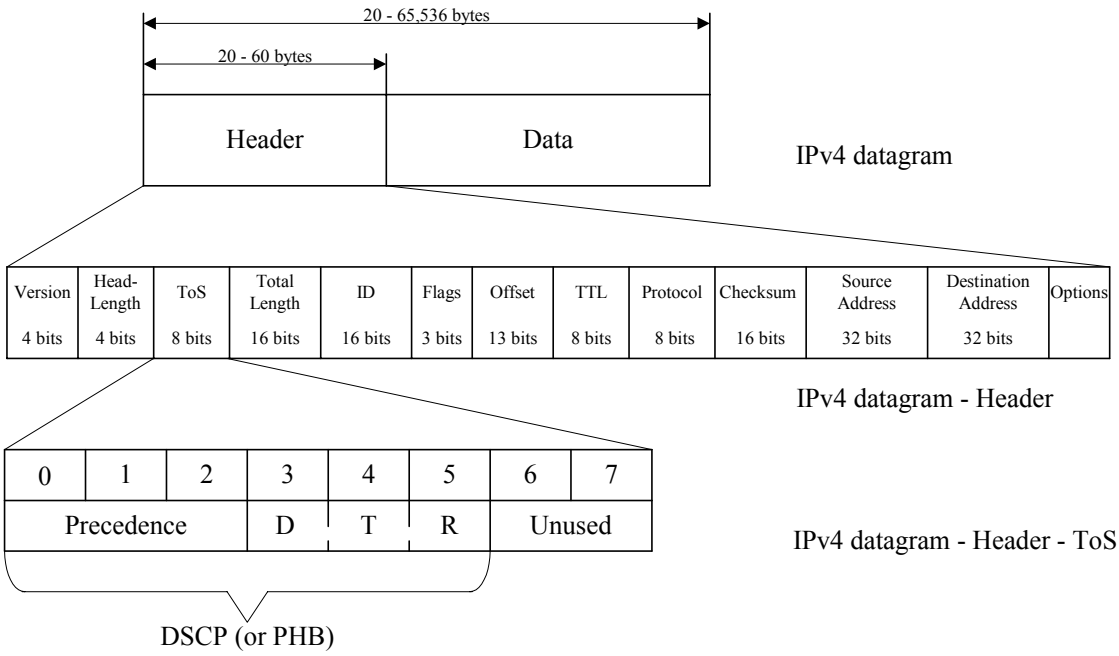


Figure 29: IPv4 Frame Structure

The 8-bit ToS field is composed of the following sub-fields:

1. Precedence: 3-bit number from 0 (“normal” priority) to 7 (“highest” priority). Precedence permits the users to indicate the importance of their packet.
2. D: 1-bit flag indicating whether the source would like low delay (0 is defined as “normal” delay; 1 is defined as “low” delay).
3. T: 1-bit flag indicating whether the source would like “normal” throughput (0) or “high” throughput (1).
4. R: 1-bit flag indicating whether the source would like “normal” reliability (0) or “high”.

The two last bits are unused and set to 0 and they are being used for congestion control. Although the precedence portion is somewhat useful, D, T, and R are not very useful.

The above preferences desired by the user are only a guide to the routers for how they

should handle the packet; they are not bound to the settings. Furthermore, since the semantics of the ToS byte were never precisely defined, a number of incompatible implementations arose. However, ToS is currently receiving a lot of attention because it is now used as one of the proposed methods for signaling the desired QoS in the Internet.

ToS information allows routing algorithms to choose different routes for IP packets based on the value in their ToS field. Instead of assigning a single metric to a link, it is possible to assign different metrics depending on the ToS value of the data. For example, if we had a link in our network that was very good for delay-sensitive traffic, we could give it a low metric for the ToS value representing low delay, and a high metric for everything else. Routing algorithm would then pick a different path for those packets that had their ToS field set to that value.

When implemented, ToS allows IP datagrams to be distinguished from each other and handled differently in times of overload. For example, when the network is overloaded, it would be useful to be able to distinguish network control datagrams, e.g. ICMP datagrams, from datagrams carrying data, e.g. HTTP messages; it would also be useful to distinguish real-time datagrams, e.g. VoIP (voice over IP), from non-real-time traffic, e.g. FTP. Cisco routers interpret the first three ToS bits as defining differentiated levels of service that can be provided by the router.

IP QoS may require retooling many of the older routers in the network. These routers are software-based routers, while the new generation of routers handles packet forwarding

via high-performance ASICs (Application Specific Integrated Circuits). Packet classification and packet scheduling are such computationally intensive functions that require ASICs-based implementation.

The IETF has redefined QoS for IP networks by proposing three schemes: IntServ, Diffserv, and MPLS.

This research uses Diffserv model for quality of service in the interconnected campus network. For one of our scenarios, the core network routers will be configure as ToS enabled devices with WFQ as the queuing algorithm, and thus capable to selectively queue the incoming packets according to the type of service they were assigned by the source application. The applications selected are database queries for reasons of suggesting a method of introducing a low-cost QoS for distributed knowledge networks. This method is implemented by assigning a higher ToS value for database queries, meaning that users querying database are given higher preference than those Web browsing, or those sending electronic mails, or those transferring files across the network. The idea is to better allocate the network resources - assuming that knowledge are stored in databases connected to the network - to activities of knowledge searching, gathering, and dispersing among the network's users. One possible extension of this approach would be the routing of these queries to specific databases depending on the domain of the queries. This is precisely the core of the second part of this thesis, when we introduce intelligence to the network enabling it to accomplish the task of routing queries to correct databases where the answer to these queries are located.

7.1.1 IntServ Model

In early 90's, the Integrated Service Model (IntServ) was proposed which provides an integrated infrastructure to handle conventional Internet applications and those QoS-sensitive applications together [7, 13]. IntServ is a fine-grained approach for QoS that uses resource ReSerVation Protocol (RSVP) as its signaling protocol. [4, 5, 16] Although IntServ / RSVP can provide QoS guarantees to applications, it has a scalability problem since each router in the model has to keep track of individual flows.

7.1.2 Diffserv Model

To address the scalability issue, a new core stateless model, called Differentiated Service Model (Diffserv) was proposed and has become a popular research topic as a low-cost coarse-grained method to bring QoS to today's Internet, especially in the backbone networks [10, 6]. The main purpose of the Diffserv model is to provision end-to-end QoS guarantees by using service differentiations in IP networks. Unlike the IntServ model, it does not keep soft states for individual flows; instead, it achieves QoS guarantees by a low-cost method - aggregating individual flows into several service classes. Therefore, the Diffserv model has a good scalability, and it is the most widely accepted QoS mechanism in enterprise networks.

The Diffserv model works as follows. Incoming packets are classified and marked into different classes, using so-called Differentiated Services Code Point (DSCP) [8] (e.g., IPv4 ToS bits or IPv6 Traffic Class bits in an IP header). Complex traffic conditioning such as classification, marking, shaping, policing are pushed to network edge routers or

hosts, which will not be considered in this research. Rather, the core routers in our prototype of an interconnected campus network will classify packets and forward them using corresponding Per-Hop Behaviors (PHBs) codes.

From the administrative point of view, a Diffserv network can be managed as consisting of multiple DS domains. To achieve end-to-end QoS guarantees, the negotiation and agreement between these DS domains are needed. Thus, packets belonging to the same DS domain are packets with the same Diffserv classification.

Therefore, in this research, packets that carry database queries are classified with a ToS(3) of "excellent" delay, throughput, and reliability, while the other applications (e-mail, http, and ftp) are given a ToS(0) of "best effort".

As shown in figure 37 above, the first 6 bits of the ToS field are the Diffserv field. The remaining 2 bits are reserved for an ECN (explicit congestion notification) field. The ECN part of ToS is for traffic control. One of the bits indicates "congestion has been experienced", and the other bit indicates whether the end nodes understand how to react to that bit; the idea is that if they don't, the router should drop the packet rather than set the congestion-experienced bit.

Diffserv does not provide QoS on the basis of flows but rather on the basis of the needs of different group of users. This means that all IP traffic is split into groups of different QoS requirements and that routers recognize different groups on the basis of a label in the IP header.

Several key characteristic of Diffserv contribute to its efficiency and ease of deployment:

- IP packets are labeled for differing QoS treatment using the 6-bit DS field in the IPv4 header. No change is required to IP.
- A service-level agreement (SLA) is established between the service provider (internet domain) and the customer prior to the use of Diffserv. This avoids the need to incorporate Diffserv mechanisms in applications. Thus, existing applications need not be modified to use Diffserv.
- Diffserv provides a built-in aggregation mechanism. All traffic with the same Diffserv octet is treated the same by the network service. This provides for good scaling to larger networks and traffic loads.
- Diffserv is implemented in individual routers by queuing and forwarding packets based on the Diffserv octet. Routers deal with each packet individually and do not have to save state information on packet flows. Also, recommendation for queuing discipline requires that routers should implement precedence-ordered queue service. Precedence-ordered queue service means that when a packet is selected for output on a link, the packet of highest precedence that has been queued for that link is sent.

The queuing discipline used is very important for the DS model. The differentiation is achieved here. First, incoming packets have to be classified according to their service classes. Then the packets must be scheduled for transmission by being assigned the appropriate queues in memory. Different scheduling algorithms can be used. These

include the following: FIFO, PQ, CBQ, and WFQ. The routers in our campus networks use WFQ as the queuing scheme.

Although intensive research has been done on Diffserv, it is still hard to progress in the research of this topic with respect to the overall impact on IP networks because it is too expensive and still not possible to deploy real Diffserv enabled routers into all of the IP networks in existence. Thus simulation is the best way to study Diffserv before deploying it to real IP networks.

7.1.3 MPLS Model

The whole idea of differentiate services for IP traffic is to allow routers to make a forwarding decision based on small header with small addresses rather than larger IP headers. It is prohibitively expensive to build a really fast router that makes forwarding decisions based on large packet header. The IETF workgroups on routing schemes are discussing the implementation of what is called as tag switching, also known as MPLS (Multi-Protocol Label Switching). MPLS is a scheme that looks like having an ATM header on your IP packet. MPLS takes all the rest of the ideas of ATM but uses IP-sized packets.

The protocols being developed by the MPLS working group are currently subject to change. There are no RFCs, and only a lot of Internet drafts. But the group seems to be converging on the basic idea of having a backbone those routes according to “labels” rather than IP headers. So an input port with a label would match up with the proper output port and label.

8. Network Performance

After availability, the second most important characteristic of a network is performance. For interactive applications, a response time suited to the application is critical for satisfactory productivity. Many applications that are not necessarily sensitive to response time, such as file transfer, disk backup, and bulk printer output, are sensitive to throughput; that is, the amount of data to be transferred is so great that a steady high throughput is necessary. Thus, network performance is centered around collecting data on these relevant variables such as response time, throughput, and utilization for further analysis.

The purpose of any network performance monitoring systems is to continuously measure and analyze the most important communications-related parameters in a network.

OPNET is the network monitoring system used in this work. A data collection module gathers statistics on raw network traffic as well as data link and packet traffic. To interpret traffic, data collection module must be programmed to recognize the data link and network protocols.

The indicators, also known as metrics, used in this paper are: network delay, throughput, and utilization. The measurements of these indicators will be taken between nodes of the network, circuits between the nodes, and at the global level - the network as a whole.

8.1 Network Metric Fundamentals

An actual reading of a campus network data can be done by easy-to-install application softwares, such as ping, traceroute, and test traffic reflectors. However, for our hypothetical campus network we rely instead on OPNET which generates traffic and provides global or local traffic statistics. The difference global and local statistics is that for local statistics, the data source for the statistics output vector is a particular module, whereas for global statistics, all the modules in the particular layer contribute to the same output vector.

Also, global statistics can be used to gather information about the network as a whole, whereas the local statistics collect information from individual nodes in the network. For example, you can find out the delay for the entire network by collecting the global delay statistic attribute or select the delay for one particular node of the network. Campus networks are generally modeled as an interconnection of a variety of elements, including links (circuits), queues or First-In-First-Out (FIFO) buffers, delay elements, servers and traffic generators.

This research will analyze the global and local network's performance in terms of utilization, throughput, queue delay, response time, and end-to-end delay.
After simulation has been carried out, the results will be post-processed to isolate the above metrics. These metrics will be analyzed to examine how they relate to one another in setting the observed network's performance.

8.1.1 Circuits (Links)

There are many widely available programs currently used to measure link performance and usage on discrete network links. Each of these tools concentrates on a particular aspect of link performance such as link latency, IP routing, best throughput attainable, or bit usage of the link. The most common measure of link performance is throughput - the second (delay being the first) fundamental property of networks that can be measured quantitatively. Combinations of digital lines and fiber optic cables interconnect all of the nodes in our domestic campus networks as well as for the international campus networks. All the links in our campus network are full-duplex links and use point-to-point protocol (PPP).

8.1.2 Network Utilization

The efficiency of a network link can be measured by link utilization which is defined as the bandwidth of a link in use divided by the total bandwidth of the link, i.e., link throughput/link data rate. Link utilization is also a measure of the cost effectiveness of a link within a network or Virtual Private Network (VPN).

The possibility of full utilization in the network under a given traffic load is an important criterion in network design. For varying traffic patterns, we use the network level analysis to determine the corresponding utilization factor for alternate network designs. This type of analysis describes the load handling capability of the campus network. The campus network utilization analysis can influence the network design choice.

8.1.3 Throughput

Throughput, also called bandwidth, is generally defined as the average rate at which packets (bits) are received at the destination, even in the event of network congestion. However, this measure of network performance has also its controversy regarding its actual definition. To many of us, throughput defines the number of correct bits, blocks, or frames received which are error-free. To others, it means the number of useful bits received that are not in error. Unless we concretely define "useful bits", it is a meaningless measure of network performance. For example, should the start and stop bits be included as useful bits? How about the header bits that directs traffic to a particular destination and carry out other control functions, but carry no useful information for the data end user? Nonetheless, OPNET measures throughput as any bit received without any error.

While one can talk about the throughput of the network as a whole, sometimes one wants to be more precise, focusing, for example, on the throughput of a single physical link or of a logical process-to-process channel.

There is a direct relationship between throughput and delay. Although, it seems that increasing throughput means reducing delay, this is not actually the case. Increasing the number of packets in the network also increases the length of the queues at each router. Longer queues, in turn, mean packets are delayed longer in the network. And that's another good reason for having a smart queuing scheme, which led this author to select WFQ as the queuing scheme for the routers.

8.1.4 Queue Delay

Packet switches generally need to store packets for some time before forwarding them on an outbound link. Queue delay is the time elapsed from the moment a packet comes into a switch (routers) to the moment it leaves it. Sometimes, a switch receives packets faster than the outbound link can accommodate them. In this case, the switch is forced to buffer these packets in its memory. Should a switch receive packets faster than it can send them for an extended period of time, then the switch will eventually run out of buffer space, and some packets will have to be dropped.

OPNET defines queue delay as the average time a packet is delayed in the switch's buffer.

8.1.5 Response Time

Response time is associated directly with application response time. It is the interval of time that spans from the application activation time to the time the data generated by the application is received at a destination. Application response time measures the end-to-end response time: the processing time it takes to starting a particular application plus the time to creating the corresponding data plus the time to transmitting this data across the link, and plus the time the data is received at some destination.

8.1.6 End-to-End Delay

The first important property of networks that can be measured quantitatively is delay. The delay of a network specifies how long it takes for a bit of data to travel across the

network from one computer to another; delay is measured in seconds or fractions of seconds.

Delays are characterized as:

- Propagation delays - delay in a network arises because a signal requires a small amount of time to travel across a wire or optical fiber
- Switching delays - an electronic device (e.g., hub, bridge, or packet switch) waits until all the bits of a packet have arrived, and then takes a small amount of time to choose the next hop before sending a packet. Fast CPUs and special-purpose hardware have made switching delays among the least significant.
- Access delays – because most LANs use shared media, computers must delay until the medium is available. Access delays are not usually large.
- Queuing delays – delays that occur in a packet switched WAN. Each packet switch enqueues incoming packets as part of the store-and-forward process. If the queue already contains packets, the new packet may need to wait while the CPU forwards packets that arrived earlier.

Network managers only care about the total delay, end-to-end delay, of a network. End-to-end delay is defined as the average end-to-end delay of data packets received at the destinations. End-to-end delay is composed of queuing delay (waiting time), which is highly load-dependent, and of transmission delay at the link.

In order to obtain simulation results that yield the upper bounds for the campus network performance criteria certain parameters that introduce delay or reduce throughput are

omitted or are set to zero. These parameters are packet switching delay, packetizing, and propagation delays, frame minimum/maximum and overhead, frame error probability. It is fully acknowledged that in real systems these parameters are not omitted and contribute, albeit negatively, to the overall performance of the network. Therefore, in order to investigate the behavior of a network under the "best" conditions and find its limitations, it was necessary to neutralize the parameters listed above.

Real-time traffic, for example multimedia distribution, requires little end-to-end delay in order to meet quality of service requirements. End-to-end delay is also called latency, which may be defined as the time that elapses from when one node sends a data to when the other node receives the data.

Experience has shown that sub second response time is necessary for reasonable productivity with interactive applications. Multimedia distribution, e.g., voice and video, is delay-sensitive traffic.

Various research results show that an end-to-end delay of more than 250 milliseconds is unacceptable to users because under this condition signal disturbance(s) become a significant problem. The ITU-T G.713 recommendation stipulates 150 milliseconds of one-way delay, but network designers strive to achieve 200 milliseconds and never exceeding 250 milliseconds of end-to-end delay.

For example, if we are requesting a 1 KB Web page, we need a data rate of 100 Kbps to transmit the page quickly enough (we employ the commonly used notation of lowercase

b for bit, and uppercase B for byte throughout this paper.). If the page contains 1 MB high-resolution images, we need a data rate of 100 Mbps.

Compare this with the fact that it is hard to carry on a conversation if the time between when you speak and when your listener hears you is more than 300 milliseconds. Thus, what we want from the network in this case is a guarantee that all our data will arrive within 300 ms. If the data arrives early, we buffer it until its correct playback time. If it arrives late, we have no use for it and must discard it.

A typical LAN used within a single building has a delay of a millisecond. Although such delays seem irrelevant to a human, a modern computer can execute over one hundred thousand instructions in a millisecond.

Therefore, low inter-application delay motivates the need for high-bandwidth paths between the applications.

Our simulation statistics include end-to-end delay, TCP delay, Ethernet delay, and response times for our applications: database query, Web browsing, email, and file transfer. They are collected at the global level as opposed to object level.
At the object level, the simulation collects queuing delay at the link connecting two of the nodes (routers), link utilization, and link throughput.

9. The Interconnected Campus Network (ICAN)

The study will model a simple, secure, and fairly representative of a typical campus network, a switch-based network, with a few standard components and uses them as the baseline for our national and global campus networks. Due to restriction on our computing resources, we scale our model to a small-to-medium size campus network. However, this model counts on its scalability and modularity to be replicated in order to represent a large size campus network, one that at any given time support the access of tens of thousands users. Our model's simplicity results in a design that is more easily managed.

The design of our campus network, as with any other campus network, has to take into consideration three factors: technology, traffic, and costs. In selecting the underlying technology of the client and server computers, networking devices, and the circuits themselves, the similarity of the design of most of the campus networks suggests the adoption of a switch-based network interconnected with other campus networks through fast routers and/or switches. Also included are fast links and traffic modeled in accordance with the type of data it represents. The growth in network traffic is immense. Today, most network designers use a 2- to 3-year planning horizon, and they also build more capacity then they expect to need – over-provisioning the networks. The shift in network costs from hardware costs to personnel costs force network designers to reduce the staff time needed to operate the network. It emphasizes the importance of the reliability of the chosen network devices when implementing the network design. Network costs will not be included in this thesis.

9.1 Geographic Scope

Common practice in the design of a network is to break the network into three conceptual parts: the access layer, the distribution layer, and the core layer. The access layer is the technology that is closest to the user and it is often a LAN or a dial-up connection over a MAN. The distribution layer connects the access layer to the rest of the network, such as the backbone networks (BNs) in a specific building. The core layer is the innermost part of the network, the part that connects the different distribution-layer networks to each other, such as the primary BN on a campus or a set of MAN or WAN circuits connecting different offices together. The core layer is the busiest, the most important part of the network. Our approach, following the building block approach, is to start with the highest level, by drawing an interconnected WAN campus network diagram, figure 30, consisting initially of six nodes: New York, Miami, Texas, California, Washington, and Illinois.

Figure 30: National Campus Network

The access network is a switch-based LAN, the distribution networks are routers/switches, and the interconnected core network will take on the circuit technologies such as digital point-to-point T-3 lines and optical point-to-point OC-x (x = 3, 12, and 48) lines. Next, we extend the national campus network (NCN) by linking international nodes to it thus forming the international campus network (ICN), figure 31.

Figure 31: International Campus Network

The nodes added are: Europe (England, France, and Germany), Asia (Japan), and Latin America (Brasil). Computing resources limited the number of nodes that could allow us to complete the simulation within a reasonable time frame.

9.2 Modeling and Simulation of an interconnected campus network

In the subsequent sections of thesis we will first describe the methodology used, next we describe the model used to simulate a typical interconnected campus network configuration. We provide a detailed discussion of the various simulation scenarios and the different process models developed and implemented as a part of these scenarios. Finally, we investigate certain system parameters (application traffic models, key protocol attribute settings and performance metrics) considered for the various simulations.

9.3. Methodology used to simulate interconnected campus network

The methodology used in this research consists of:

- The suggested interconnected campus network has typical switch-network nodes connected via different bandwidth links, from low (45Mbps) to high (2.5Gbps). Traffic will result from typical in-campus user applications: email, file transfer, database query, and Web browsing.

- A simulation model of this campus network, which will consist of an optimal arrangement of different campus network interconnected by transmission medium that supports traffic under optimal network parameters. The modeler chosen is OPNET which is discrete event simulator: the discrete event in the models will be the transmission data packets flowing through the network. OPNET software provides a means to analyze the behavior of the campus network without the expense of building a real one.

- The modeler will permit to change the configuration through iterations for optimal results in terms of performance measures. The modeler provides an extensive library of nodes, links, protocols, and traffic objects.

- The simulator will generate the actual traffic based on a probability distribution. Network traffic refers to the messages sent between nodes, and workload refers to the internal activities of the node's processors or busses. Quality of Service (QoS) will be implemented via ToS (Type of Service) which is a field of the datagram carrying the information through the network. ToS allows for differentiation between different applications used in the network.

- The simulator allows that each source device (workstation or server) on the campus network can be designated as the data source having a bandwidth (throughput) requirement, a destination drawn from a probability-weighted list, duration drawn from a data holding time distribution, a routing class, and other attributes.

- The traffic generated by the simulator will be based on the combination of exponential interarrival time and Poisson packet size. Poisson distribution can be used to represent the way that networks packets arrive at a network node. Also, the traffic flowing through the links connecting the routers at the edge of the network will be IP traffic.

- A sensitivity test will be conducted on the network to verify the reliability of its configuration under the situation where a link is broken. In our test, we actually disable one of the distribution switches and check if the backup switch maintain the network functioning.

- `Backload traffic will be included in the network in order to study its effect on the network's performance. It is important to test the network with the allowed applications on the foreground and some background traffic (other than the one from the applications) to ascertain the performance of the network. This stress test is a good measure of the network efficiency.

- This study will also include analysis of different queuing methods: FIFO and priority queues [15, 16]. The routers, (or ATM switches) along the network can select higher priority packets for processing before lower priority packets using priority queues. The routers will be configured to apply ToS to the incoming packets by looking at the ToS field of the IP datagrams and then using Weighted Fair Queuing to batch them accordingly.

. Initially, a national interconnected campus network will be studied, and then new connections to this national network will be added to form a global network.

10. National and International Campus Networks

Both campus networks, national and international, are comprised of interconnected nodes (subnets) level 3 router hardware via very high speed data links. These routers, in turn, are connected to the campus switch-based LANs. The use of routers in the campus area network to interconnect LANs instead of low cost switches is driven by the administrative and operational control routers enable. With a standard internetwork protocol, we can carry out network interconnection at level 3, or the network layer, of the Open Systems Interconnection model. The use of routers for network interconnection at this level permits us to be independent of the physical network. A router connects two or more networks, which may be of the same or different technologies.

The national campus network (NCN) consists of six nodes placed around the country: NY, Miami, Houston, Los Angeles, Seattle, and Chicago. Each node (subnet) represents a campus network. Figure 32 shows the node layout of this interconnected national campus network with the Application, Profile, and IP QoS configurations.

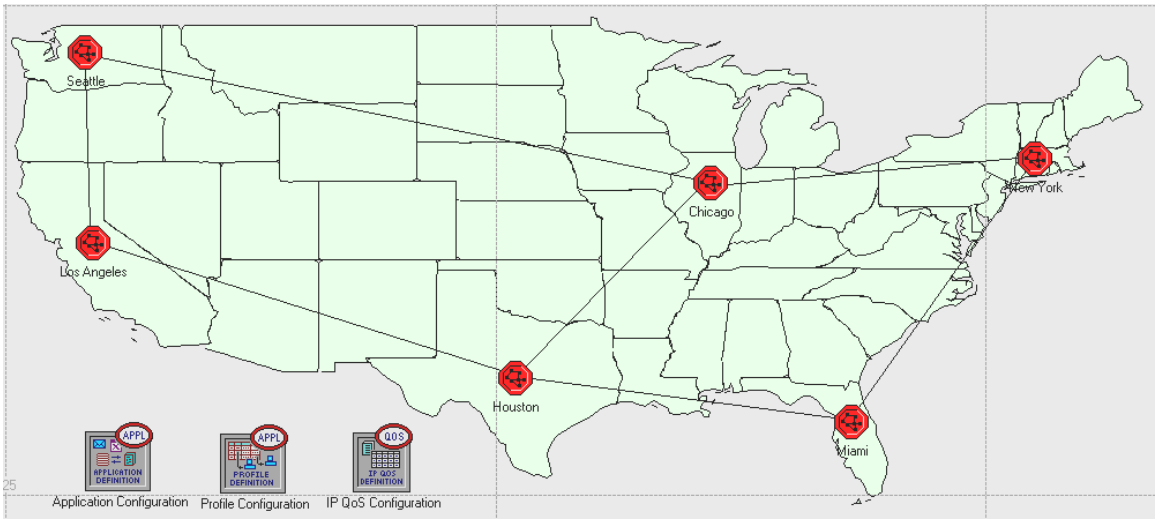


Figure 32: Node Layout of the National Campus Network

The same topology, router-interconnection with switched-based LANs, is applied to the international campus network (ICN). This global campus network consists of links from these domestic nodes to campus located in other countries.

Figure 33 shows the ICN's selected sites in Asia, Europe, and Latin America. These sites were chosen because of their strategic location and their volumes of traffic are readily available as recorded by some of Abilene's international peer networks: TransPac (Japan), Geant (Europe), and AmPath (Latin America).

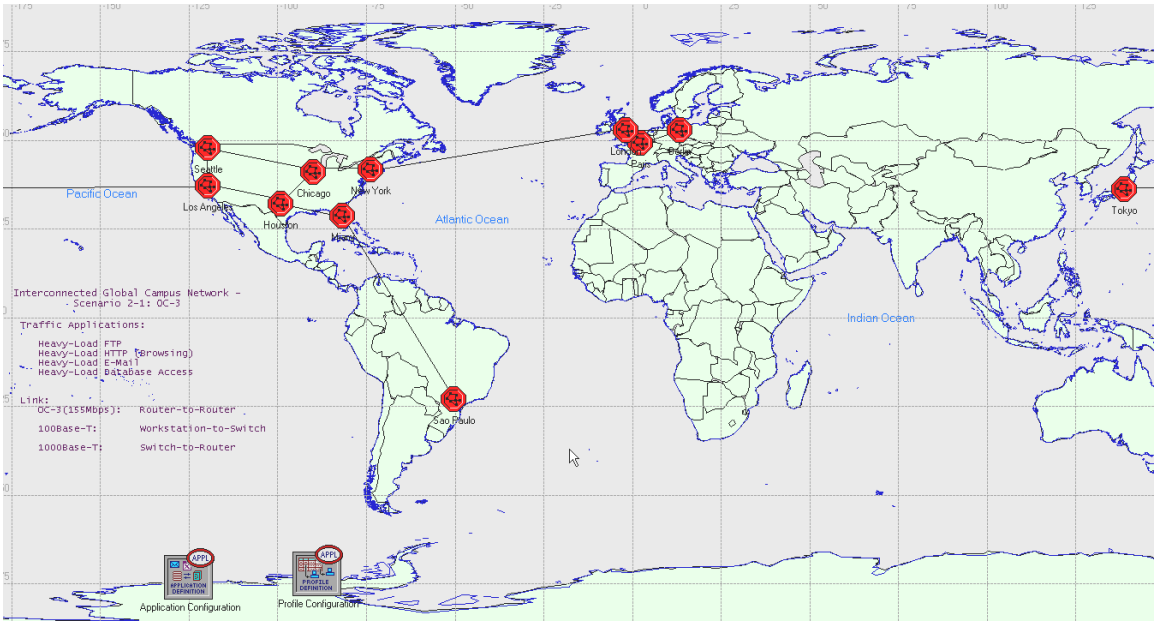


Figure 33: Node Layout of the International Campus Network

The number of nodes were restricted due to the limitation of the computing resources available. With the simulation results, we can then compare the statistics from our model with those of Abilene's peer networks available in the Abilene's web site.

These are the international nodes interconnected to our national network:

. NY to London (UK)

. Miami to São Paulo (Brasil), and

. Los Angeles to Tokyo (Japan).

These are all transatlantic circuits. In reality, they are implemented via fiber optic transatlantic cables operating at speeds 2.5Gbps on connections from US to Europe, 155 Mbps from Los Angeles to Tokyo, and at 45 Mbps from Miami to São Paulo.

In order to load our circuits to meaningful capacity and verify the interconnected network, we are also running three more connections starting out from our London node:

. London to Paris (France)

. Paris to Berlin (Germany)

. Berlin to London

10.1 Network Traffic

The flow of traffic in modern campus networks is extremely complex. The various interacting layers of protocols, for example, HTTP, TCP/IP, Ethernet, and PPP, provide a framework for the interaction of the information flows generated by a multiplicity of applications. The times when applications make requests are not scheduled, and the pattern of packets transmitted is never quite the same. The challenge of the network designer is to have enough resources (bandwidth, buffering, processing) in the right configuration to handle the traffic at any given time. Measurement and performance modeling are essential tools in carrying out this task [23].

Once the network architecture model is built, the next step is to define the network's load characteristics. There are two ways to define traffic within the network architecture: internal traffic definition and external traffic definition. Internal traffic can be generated using the traffic generators from within the framework. External traffic data represents the "real" network traffic. However, in this research, we only deal with internal data traffic, a synthetic network traffic generated by the simulator.

Network users are represented by traffic sources, which may generate the traffic for one or more simultaneous connections associated with some user activity. In order to model a given user, the traffic which that user would inject into the network must first be characterized in terms of parameters such as mean interarrival time, mean packet rate and peak packet rate. Random number generators with appropriate probability distributions are then used in the simulator to generate the event times to model the traffic. A detailed description of source modeling is beyond the scope of this thesis.

We are interested in the performance of a typical and scalable representative-design of a campus network – meant to be a blueprint defined as a switch-based network at its core with outer connecting router(s). Every interconnected campus network consists of 1000 users, which represents a small-to-medium size a campus network [24]. The NCN has six nodes – giving a total of 6000 users, and the ICN has eleven nodes – a total of 11000 users. The explicit traffic is generated using standard campus network applications: email, Web browsing, ftp, and database queries.

It is important to note the load sharing capability of the routers, that is, the capability of a router to distribute traffic over all its network ports that are the same distance from the destination address. The routers used in this project, Cisco routers, are said to have a good load sharing algorithms that use both line speed and reliability information. Also, load sharing increases the utilization of network segments, thus increasing effective network bandwidth.

Also, in order to make the network modeling feasible, the following assumptions are made about the network traffic:

- Traffic characteristics for all N users are identical.
- All N users have one or more active sessions, i.e., users are not idle (except for the interarrival time between packets).
- Applications generating traffic occur simultaneously, i.e. data traffic from any application can be sent and another type of data traffic application can be received at the same time.
- The traffic characteristics of each user should not change for different work loads. Instead, loading is changed by changing the number of users.

We simulate bursty network traffic using OPNET. Bursty traffic involves data being transmitted in bursts with idle periods of arbitrary durations in between data transmissions. TCP is the transport protocol for all the applications. The model of TCP used in the simulations is based on the TCP-Reno version. It supports Jacobson's congestion control mechanism [31], exponential back-off, enhanced round trip time

(RTT) estimation based on both mean and variance of the RTT, Karn's algorithm and the fast retransmit/recovery mechanisms. The default TCP buffer size of 64 KB is used as well as the TCP maximum segment size (MSS) of 1500 bytes (i.e. IP MTGU of 1540 bytes).

This project does not include non-bursty traffic or multimedia traffic, e.g., voice and video, which involves data being sent at a fixed rate with no bursts. Since our campus network model is switch-based, multimedia traffic would become a broadcast traffic flooding the network. This is because multicast instructions reside at the protocol layer (layer 3), which the switch is unable to recognize. The switch therefore sends the traffic out all of its ports. The alternative is to configure the switch with filtering mechanism, however, this will not be considered in this research.

10.2 Subnets

Each node in the national and global campus networks represents a subnet or sub-network, which is a group of IP addresses. In each individual campus network considered for this research, an assumption has been made that the subnets used are Ethernet LANs – in reality, this is what is commonly used in real campus networks. Each subnet is assumed to be consisting of 1000 workstations with a layer 2 Ethernet switch. All workstations are connected in full duplex mode to the switch - the stations are transmitting and receiving simultaneously. A full duplex Ethernet means running Ethernet without the CSMA/CD MAC in operation.

Our basic configuration of a subnet is shown in figure 34. The New York subnet was selected for this display. It is considered to be the standard configuration in networking according to some leading companies like Cisco [24]. Redundancy is built in all levels, from the backbone to the distribution switches. In case of failure in any connection the redundancy link can take over the network operation. This level of redundancy needs to be installed to ensure that critical parts of the network operate correctly at critical times.

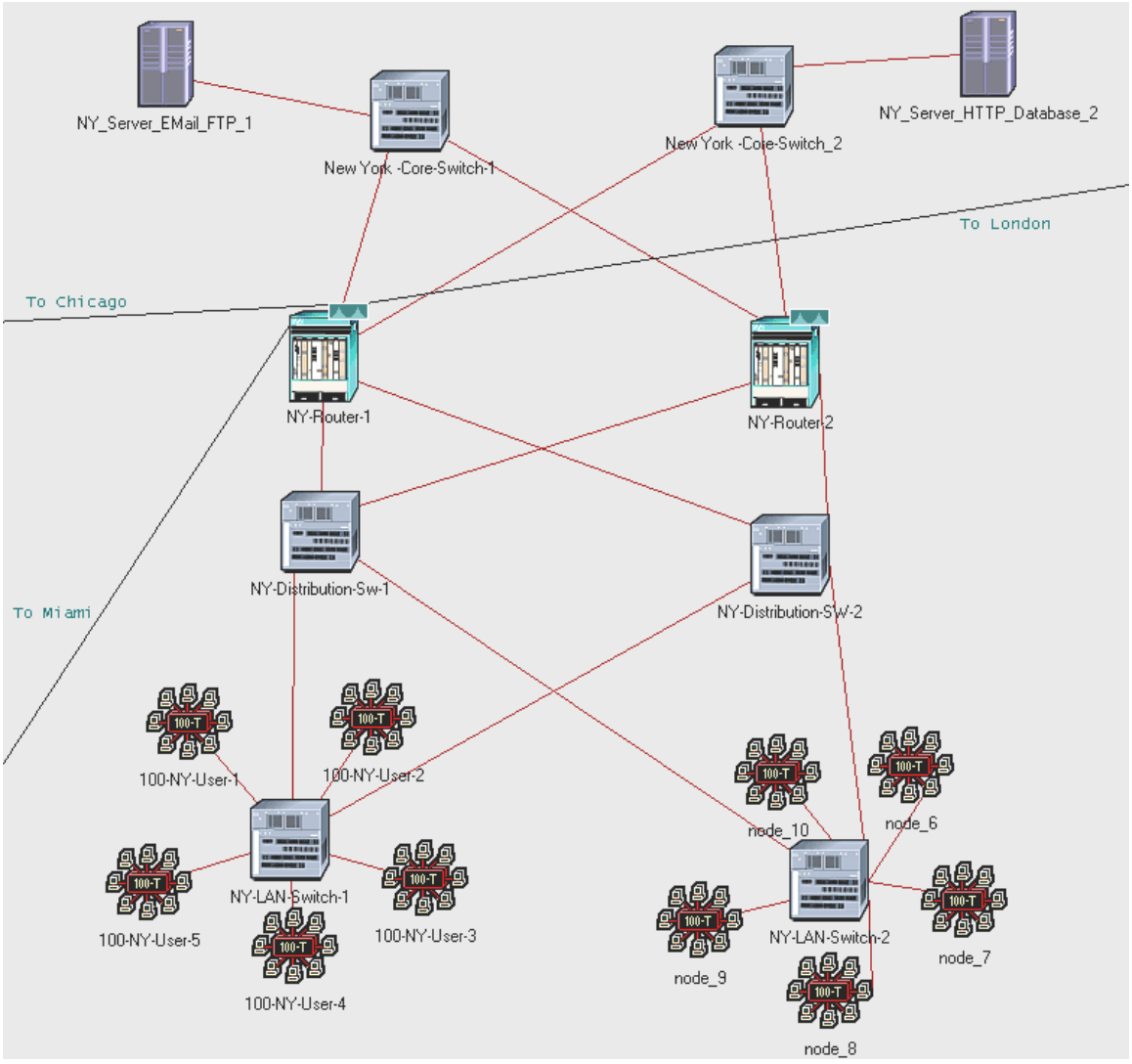


Figure 34: NY Subnet

Although redundancy increases the cost of initial setup hardware, over time, however, lower maintenance and savings in potentially lost revenue could pay for the difference in capital costs. Nonetheless, this research does not account for network costs.

10.2.1 Switch

A switch is an excellent device for reducing network congestion and for providing point-to-point connections. The LAN switches and the distribution switches are "Ethernet_128_adv" provided by OPNET Modeler library. It implements the Spanning Tree algorithm in order to ensure a loop free network topology. Packets are received and processed by the switch based on the current configuration of this Spanning Tree.

10.2.2 Router

The routers at the core are Cisco model CS_12016_16s_a10_fe8_ge3_sl24_adv. This model represents a specific configuration of an IP-based router switch model. It supports the following interconnections: 100Base-T Fast Ethernet, 1000Base-T Gigabit Ethernet, OC-3 and OC-12 ATMs, and OC-3, OC-12, OC-48, and OC-192 SONETs.
Its operation consists of routing IP packets arriving on an IP interface to the appropriate output interface based on their destination IP address. The Routing Information Protocol (RIP), Open Shortest Path First (OSPF) protocol, Border Gateway Protocol (BGP), Interior Gateway Routing Protocol (IGRP) or Enhanced Interior Gateway Routing Protocol (EIGRP) may be used to automatically and dynamically create the routing tables and select routes in an adaptive manner. The routers in our model use RIP as the default routing protocol. RIP is known as a distance vector protocol. It builds a table (or

database) in memory that contains all the routes it has learned and the distance to that network. A router associates a cost with those interfaces. As other routers report their topology and connectivity regarding locally attached subnets, all the information needed will be stored in each router's database (routing table). Routing will then be based on cost, which really equates to hop count.

The key model features for our routers are: an IP forwarding rate of 60,000,000 packets/sec, and an implementation of a "store and forward" type of switching methodology. It has an aggregate switching capacity of 80 Gbps.

10.2.3 Links

Links are used at each geographical node to connect:

. workstations with the switch,

. switch with the router, and

. switch with servers (email, Web browsing, database, and file transfer).

All the links in our network use point-to-point protocol (PPP) and are full-duplex links. These are the links used in our model:

- T-3, OC-3, OC-12, and OC-48: each is used alternatively to interconnect the subnets. Their speeds are 45Mbps, 155 Mbps, 622 Mbps, and 2.5 Gbps respectively.
- 1000Base-T: it is used to connect switches and switches with routers. It is a full-duplex link and represents an Ethernet connection operating at speed of 1Gbps.

- 100Base-T: it is used to connect the workstations with the switches and switches with the servers. It is a full-duplex link, which represents an Ethernet connection operating at a speed of 100Mbps.

The various configurations of the campus networks in this project were designed on the OPNET modeling software. Basically, in all configurations the subnet location is unchanged, while the link connections are changed from copper wire to fiber optic to evaluate the effect on performance.

10.2.4 Server

The server used in this project design is "Ethernet_server_adv" provided by OPNET's model library. The network has two servers, one hosts the file transfer and email applications, and the other the Web browsing and the database queries. For security reasons, each of the servers could have been duplicated, however the time to complete a simulation is augmented by this redundancy, and therefore it was not implemented.

10.2.5 LAN

Our subnets, see figure 40, consist of two switch-based LANs each with five 100 Ethernet workstations. These 1000 users in each subnet generate a variety of data traffic – this data can be transferred from one or more servers, as well as directed to both, the internal servers and external servers. We chose email, FTP, HTTP, and database query as the applications that run at these workstations. TCP is the transport protocol for all the applications. The switching speed: (default = 500,000pkts/sec) determines the service

time for each packet while going through the switching module of this node. All packets, i.e. incoming, outgoing and transit packets, are subject to this delay.

Additionally the incoming and outgoing packets experience transmission delay, which is based on the link data rate modeled within the node, and higher layer delays, like delays by IP and application layers, if applicable. However, as mentioned before, our end-to-end delay analysis will not take into account these sources of delay. The workstations support an underlying full-duplex Ethernet connection at 100 Mbps.

10.3 Configurations Characteristics

One crucial element to understanding network performance is to baseline the traffic patterns on the network. By baseline, we mean gathering data on the traffic flow through the network during normal operation. Once we baseline the traffic multiple times during the day, we can then understand what the network's basic performance level is. This includes response times during peak and low traffic periods, types of traffic traveling through the campus, and network utilization of different parts of the network.

The model for the campus network configurations follows our basic "small-to-medium campus size", 1000-user switch-based campus networks interconnected via routers as shown in figure 35. To give an idea of the scale of our campus network, NYC College of Technology, one of the CUNY colleges, has 10,500 students enrolled, and at any given time it has around 2,000 workstations accessing the campus network. The whole CUNY has around 30,000 workstations accessing the network at any given time.

The baseline we use is the normal operation of the network without any interference of extraneous load - those that are not the ones generated by our campus applications (Web browsing, email, file transfer, and database query). Each user is a “Heavy-user Ethernet client”, which is a modification of a typical campus user traffic scenario: light-loaded email, light-loaded FTP, medium-load database, and heavy-loaded Web browsing. The heavy-user is characterized by heavy-load e-mail, heavy-load FTP, heavy-load Web browsing, and heavy-load Database Query. The purpose is twofold: to generate a traffic load level comparable to an actual load of a similar campus network that can be simulated with the computing capability at hand, and to test the network configuration under different conditions of link, load, and protocol.

The scale of the load generated, which is not much over that of a real campus traffic load at its highest point (see figure 35), can thus allow for extrapolating the analysis to that of a similar configuration but with over 10 times the number of users of the network (see figure 36).

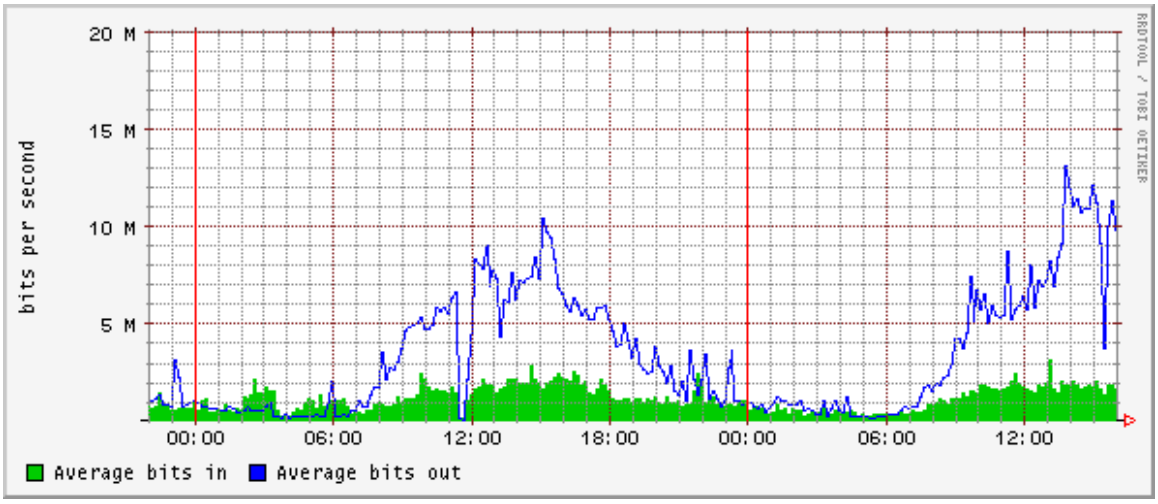


Figure 35: Daily Traffic of Brooklyn College, CUNY

A snapshot of the daily network traffic of the City University of New York (CUNY) is shown in figure 36. Note that CUNY comprehends 19 colleges and has 400,000 enrolled students, and at any given time an average of around 30,000 workstations are accessing the network. The traffic (focus on the higher plot which includes traffic from all colleges) is recorded at the central gateway router located in the university's headquarter building.

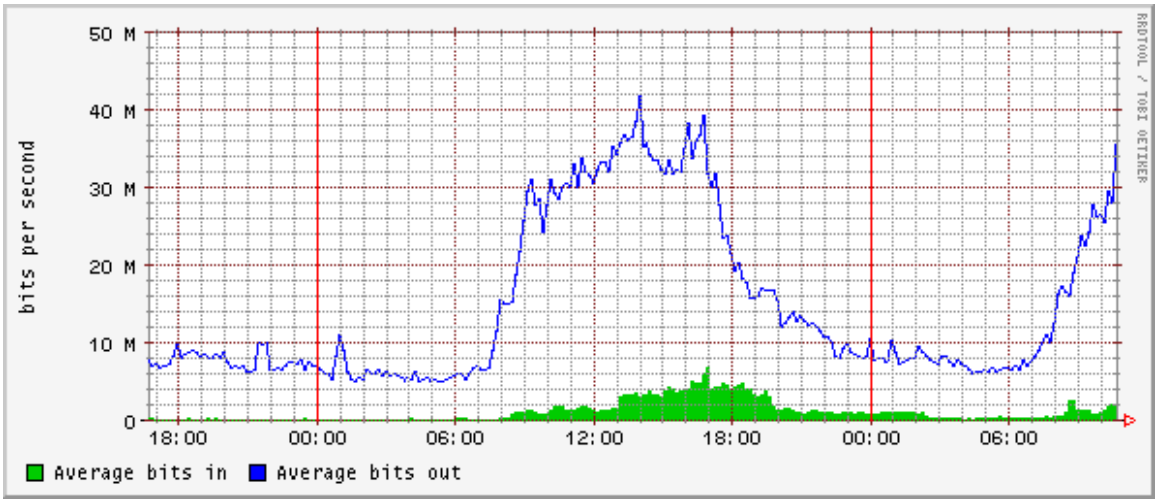


Figure 36: Daily Traffic of CUNY

10.3.1 User's Characteristics

Each Ethernet LAN station generates constant-size files at a rate that follows the Poisson distribution - the assumption is that the events follow an exponential distribution with a certain mean. Poisson model which is not appropriate for modeling at the packet level, however, it is still very useful to model application level events. FTP sessions, HTTP sessions, and database searches are statistically consistent with Poisson arrivals [27, 28].

The “Heavy-user Ethernet Client” applications parameters are shown in table 8.

Application	Load	Inter-Arrival Times (seconds)	File Size (bytes)	ToS††
E-Mail	High	Exp(360)*	Poisson(2000)	Best Effort (0)
FTP	High	Exp(360)	Poissson(50000)	Best Effort (0)
HTTP	High	Exp(60)**	Poisson(1000)†	Best Effort (0)
Database	High	Exp(12)	Poisson(32768)	Best Effort (0)

Table 8: Heavy-User Applications Parameters

* Inter-arrival times of sending emails and receiving emails

** Page inter-arrival times

† File size per page

†† ToS changes from ToS(0) to ToS(3) (“Excellent Effort”) for the scenario where IP QoS is introduced.

The start time of the next email is computed by adding the inter-arrival time to the time at which the previous email completed. The start time for a file transfer session is computed by adding the inter-request time to the time that the previous file transfer started.
The start time for a page request is computed by adding the inter-arrival time to the time that the previous page request completed. The start time of the next database transaction is computed by adding the inter-arrival time to the time at which the previous database transaction completed.

10.3.2 Applications Characteristics

Type of Service (ToS) is the sub-field of the header part of the IP datagram, which is used for quality of service in IP networks. One of the scenarios has the routers programmed to recognize differentiated traffic with a specific value of ToS. All of the applications, in this particular scenario, are classified as ToS(0) with a Best Effort delivery, except DB Query which is given ToS(3) with an Excellent Effort delivery mode.

ToS assignment at the client is not affected by the ToS value specified at the server. Some of the symbols and their respective values of ToS is shown in table 9.

ToS	Description	Value
ToS(0)	Best Effort	0
ToS(1)	Background	32
ToS(2)	Standard	64
ToS(3)	Excellent Effort	96
ToS(4)	Streaming Multimedia	128
ToS(5)	Interactive Multimedia	160
ToS(6)	Interactive Voice	192

Table 9: ToS values

ToS value is an integer between 0 - 252, 252 being the highest priority, assigned to packets sent from the client. It represents a session attribute which allows packets to be processed faster in IP queues.

10.4 Simulation Scenarios

Table 10 summarizes the scenarios with varying links, device failure, explicit and background traffic, and quality of service for both the national campus network (NCN) and the international campus network (ICN).

	Scenario11	**Scenario12**	**Scenario13**	**Scenario1F**
T-3 (45Mbps)	Link	+ Backload	+ QoS	Link Failure
OC-3 (155Mbps)	Link	+ Backload	+ QoS	
OC-12 (622Mbps)	Link	+ Backload	+ QoS	
OC-48 (2.5Gbps)	Link	+ Backload	+ QoS	

Table 10: Simulation Scenarios

These different scenarios will be analyzed according to the following metrics:

- End-To-End (ETE) delay: TCP and Ethernet
- Response times: E-Mail, FTP, HTTP, and DB Query
- Link: Queue Delay, Throughput, and Utilization.

The metrics for links are analyzed for the following subnet interconnections:

. NCN: Houston–Chicago, Los Angeles–Seattle, and New York–Miami

. ICN: Houston–Chicago, Los Angeles–Seattle, New York–Miami, New York–London, Miami–Sao Paulo, and Los Angeles–Tokyo.

10.5 Network Links

Simulation will be conducted over individual links and over a comparison between all of them. These links (T-3, OC-3, OC-12, and OC-48) were selected in order to correspond to the ones that are actually used in the Abilene network, except that OC-192 (10Gbps), which is used in the Abilene network, but it is not included in this research because the generated loads in our networks are not high enough for meaningful utilization when OC-192 link is used.

10.6 Background Traffic

Network data traffic is susceptible to delays caused by additional, unrelated traffic on the lines. We will model network performance without and with this background traffic to determine how this background traffic affects our network traffic represented by electronic messages, file transfers, web pages, and database queries. OPNET treats background traffic as an analytically modeled traffic that impacts performance of explicit traffic by inducing additional delays. This background traffic is used to define traffic behavior through the so called degradation policy, see table 11.

Background Load for 10-minute simulation	
Simulation Period (sec)	**Percentage (%)**
0 – 200	30
200 – 400	40
400 – 600	50

Table 11: Backload Traffic Levels

The percentage numbers are typical values for background component of nominal traffic, and they represent respectively a low, medium, and high level of interference (other connections using the same link) to the network data traffic. Degradation policies would specify how much of the network resources each traffic class could receive under different levels of congestion. In particular, the degradation policy would specify traffic behavior in terms of the performance parameters like delay, loss (IP packets dropped), and throughput - how loss and delay that the traffic class experiences should change due to variation in the network congestion levels.

Such an approach, degradation and QoS, to service differentiation allows for dynamic allocation of traffic resources while maintaining fairness and predictability of the traffic behavior under all network conditions.

10.7 Sensitivity Test

Interface failure analysis, also known as sensitivity test, is commonly used by network designers. From a practical point of view, sensitivity testing involves breaking an active and stable link(s) and observing how the change is handled by the internetwork: how traffic is rerouted, the speed of convergence, whether any connectivity is lost, and whether problems arise in handling specific types of traffic. We selected one of the distribution switches to fail.

Bear in mind that the switch failure will put strain in the remaining interfaces, and might force the network to slow down.

10.8 Run-Time

Our simulations run for 600 seconds, 10 minutes, and all the applications that generate the traffic (database queries, Web browsing, electronic mails, and file transfers) start simultaneously with a starting time following an uniform distribution between the first 100 to 110 seconds of simulated time, that is, every event has the same probability to occur at every value on or between 100 and 110 seconds. The first 100 seconds is the startup time, aka warm up time, for the simulator to reach stability in generating the traffic. Data are then generated according to an exponential distribution until the end of the simulation time.

Again, due to limitations of our computing resources, we could not run our simulations within a reasonable time for more than 10 minutes, especially for the global configuration where it takes over 22 hours of computing time to run a 10-minute simulation.

The simulation is implemented in OPNET Modeler 10.0.L running on a Toshiba laptop with Windows XP, Pentium IV 1.7 GHz with 1GB of RAM. Simulation is run over a 10 minutes period.

11. Simulation Results

It took in average 8 hours to finish ten minutes of simulation for the national network configuration of six nodes spread out across the country, and double the time to simulate the same 10 minutes for the international campus network of 11 nodes.

The distributions of the parameters (ETE delay, Queue delay, Response Times, Throughput, and Utilization) are plotted as time average values in order to produce a mean of the results over time. Before showing the simulation results for different scenarios, figure 37 shows the national campus network traffic load generated by e-mail, ftp, db query, and http.

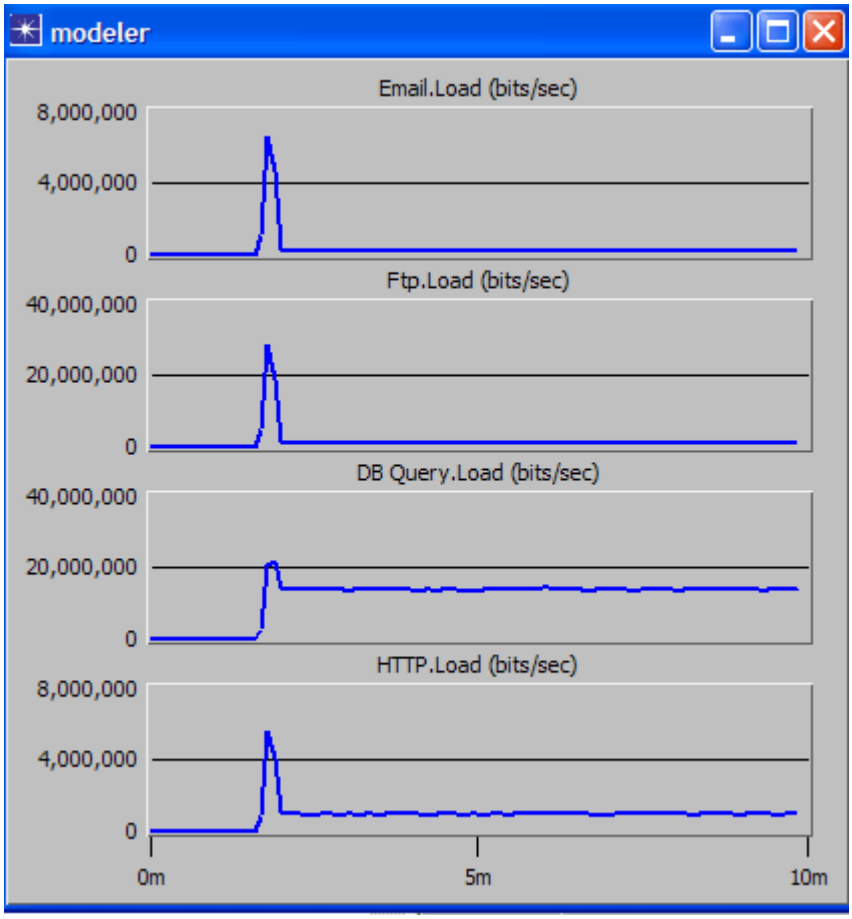


Figure 37: NCN: Traffic load comparison

The peaks we see in figure 37 indicate the time, between 100 and 110 seconds, when the data traffic starts to flow through the links of the network. After the peak the traffic load is almost constant for every application. The sample mean load is 0.3Mb for e-mail,

1.5Mb for FTP, 0.9Mb for HTTP, and 13.5Mb for DB Query, adding up to the total of 16 Mb/s which is comparable to that of Brooklyn College, see figure 37. Table 12 shows a more detailed traffic load (bits/sec) of each application.

Statistic	Mean	Maximum	Minimum	Standard deviation
E-Mail	281,087	6,181,461	0	756,000
FTP	1,192,586	26,322,493	0	3,190,385
DB Query	13,534,756	21,561,003	0	5,420,623
HTTP	851,162	4,611,388	0	616,766

Table 12: NCN: Traffic Load

The results shown in table 12 can be easily verified; for example, according to table 8, HTTP has a packet size of 1000 bytes and interarrival time of 60 seconds, which means 133 bits/sec. The network has 6000 users. Therefore, the HTTP traffic load results around what is shown in table 12.

11.1 National Campus – Scenario 11 – T-3, OC-3, OC-12, OC-48

The national campus network (NCN) consists of six subnets placed around the USA interconnected via full-duplex links through the core subnet routers.

Each simulation runs for a different link (T-3, OC-3, OC-12, and OC-48) for 10 minutes.

The connection between workstations and switch as well as the connection between switch and the servers are 100Base-T full-duplex links.

A gigabit Ethernet link connects the switches to the routers.

The user profile is our “Heavy User Ethernet Client”, as it is with all the configurations.

11.1.1 NCN: End-To-End (ETE) Delay - links

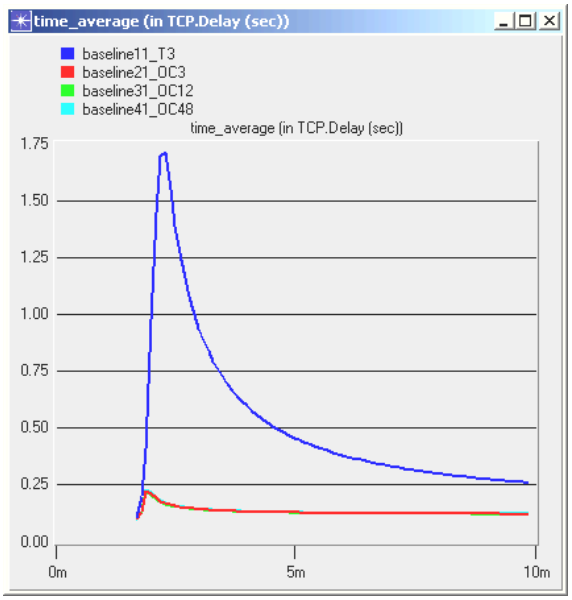


Figure38: NCN: TCP delay (sec)

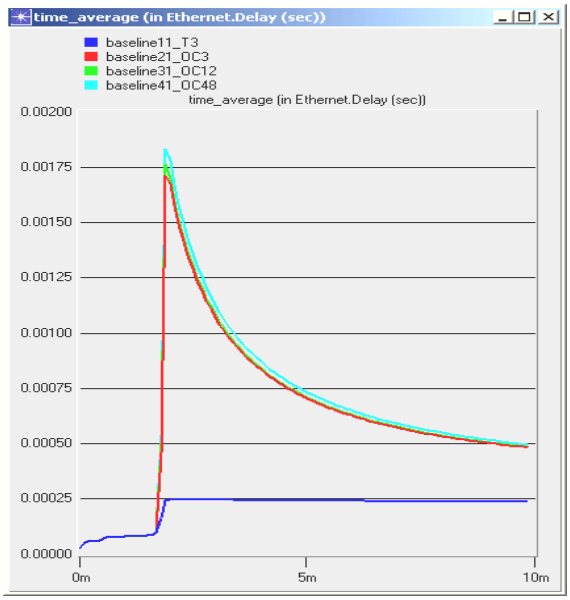


Figure 39: NCN: Ethernet delay (sec)

. Our applications, e-mail, FTP, HTTP, and DB Query, are the traffic in our networks. The transport protocol for these applications is the TCP which is regarded as a slow but reliable protocol. TCP delay is an overall average, for all applications being used in the network, of the delay of all the packets belonging to a specific application session. The TCP delay is also augmented by the retransmission time of the acknowledgements. The way that TCP works is that it keeps increasing the sending rate of packets as long as no packets are lost, and when they do start being lost or dropped, TCP rearranges the sender's sliding window. TCP delay is inversely proportional to the bandwidth. At the lower end of the TCP delay graph, when the network reaches steady state, where values become constant, TCP delay reaches a value of 0.26 seconds for T-3 and 0.13 seconds for all of the optical links, which is exactly half of that for the digital link.

. The Ethernet delay is defined as the end to end delay of all packets received by all the stations and represents a global performance of the network. Network latency depends on load, drop link speed, packet sizes, architecture and the number of switches between the server and the client. The switch latency may vary from a few tens of microseconds up to several milliseconds, and it may also vary due to the switch load. Most new Ethernet switches are based on 'store-and-forward' technology. This means that a packet must be completely received on an input port before it is checked for bit errors and compared with the FCS found at the end of the packet and forwarded to the respective port. Thus, the latency depends on the drop link speed and the packet sizes. The graph of ETE delay shows a reverse situation of that of TCP delay; i.e. the delay for T3 is half of that for the optical cables which have the delay at 0.5 milliseconds. For most applications, the latency (and end-to-end transit delay) of switched-LANs is low enough to have negligible impact on the response time and performance perceived by the user. This is because latency of less than 250 microseconds or so is typically very small compared to the overall end-to-end system latency (compare that with TCP delay of 250,000 microseconds), which can accumulate to many milliseconds when application queuing delays, receiver buffering/interrupt delays and wire delays are all considered. The faster optical cables deliver the packets faster causing congestion in the need to buffer the packets in the switch, and thus higher delay. Our results shows Ethernet delay within the acceptable range of less than 500 microseconds for the digital and the optical cables.

11.1.2 NCN: Response Time - links

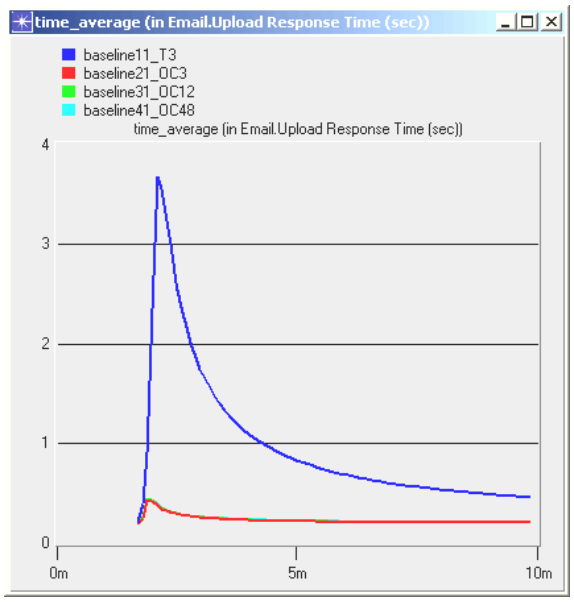


Figure 40: NCN: E-Mail response time

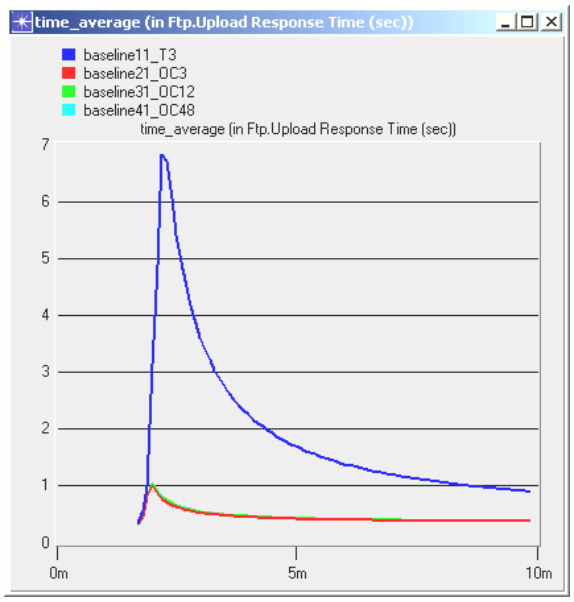


Figure 41: NCN: FTP response time

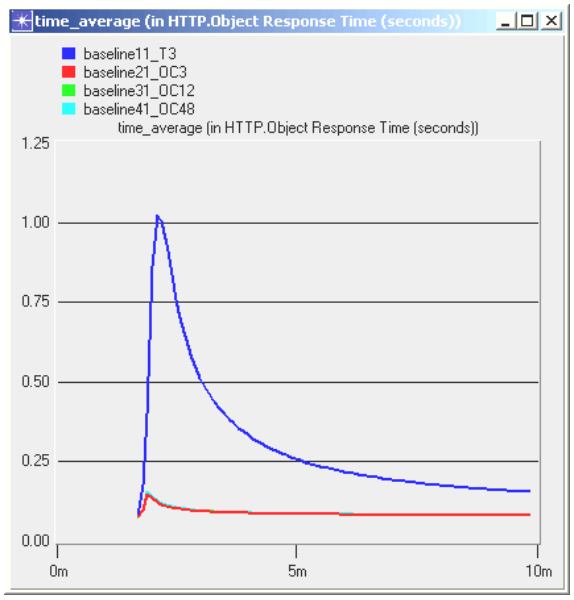


Figure 42: NCN: HTTP response time

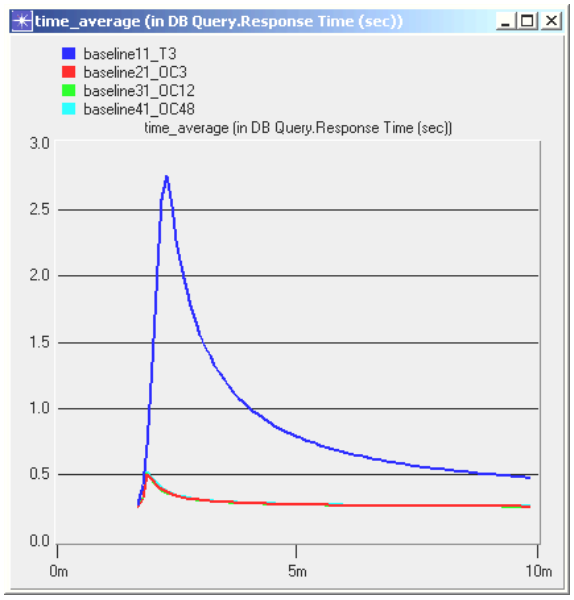


Figure 43: NCN: DB Query response time

Response time is measured from the time a client application sends a request to the server to the time it receives a response packet. The graphs for response times for the applications without T-3 shows no difference from what the figures above is showing. The response times for every application are the same for OC-3, OC-12, and OC-48.

The graphs for upload and download response times are the same. The relationship between an application response time and the links are the same as the one between TCP delay and the links, that is, the faster optical links present a response time half of the duration of that of the T3 link.

Again, the peaks in the beginning of every graph represent the interval of time during which the applications are activated and thus the data traffic starts to flow through the network. At the tail end of the distributions, the mean values (in seconds) are shown in table 13.

	T-3	OC-x
E-Mail	0.52	0.24
FTP	0.90	0.40
HTTP	0.16	0.08
DB Query	0.47	0.26

Table 13: Mean Response Times

Page contains an object (1000 bytes in size) and five medium-size images (between 500 to 2000 bytes each). Therefore, object response time is lower than page response time.

11.1.3 NCN: Queue Delay - Links

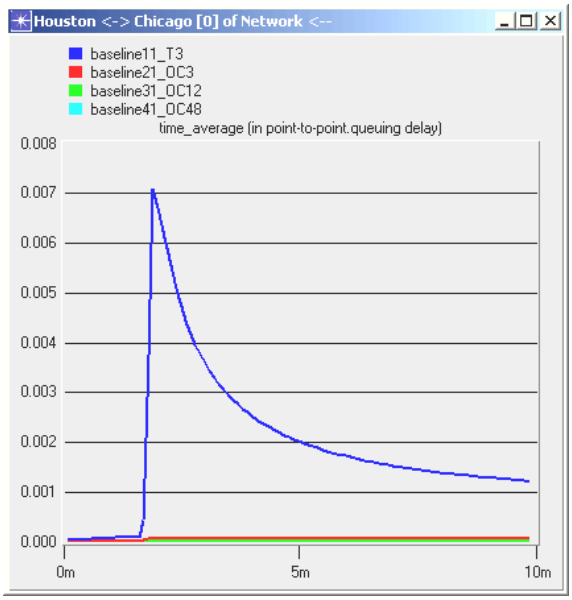


Figure 44: NCN: Queue delay Hou-Chi

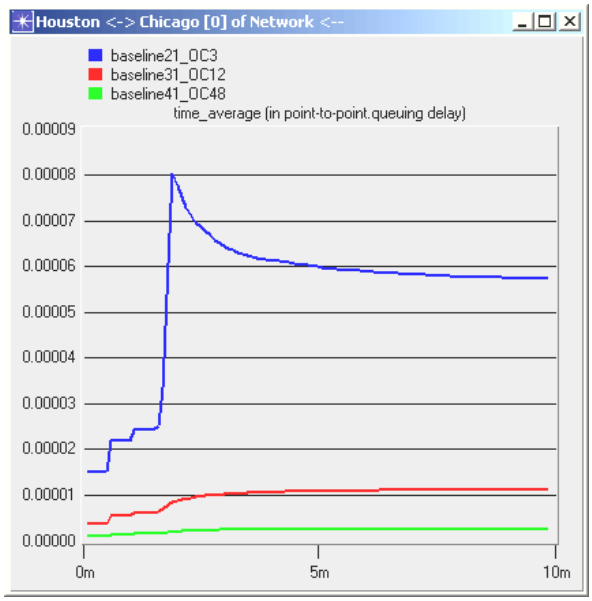


Figure 45: NCN: Queue delay Hou-Chi no T-3

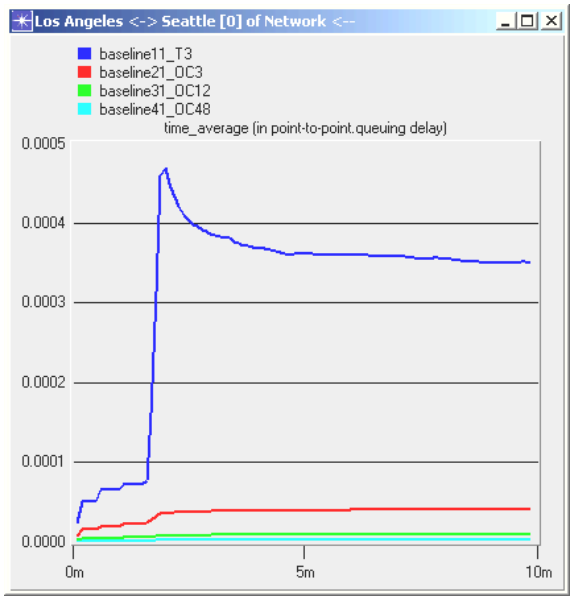


Figure 46: NCN: Queue delay La-Seattle

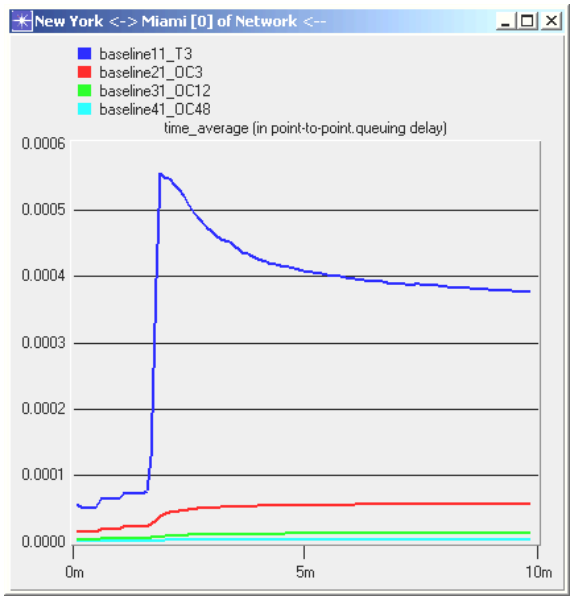


Figure 47: NCN: Queue delay NY-Miami

For our load, T3 link skews the distribution for the faster optical links, and that's the reason why for a higher load Houston-Chicago the optical link distributions had to be shown separately from the T3 distribution.

The mean queuing graphs show the delay experienced by the packets in the queue during the simulation.

The large change early in the simulation reflects the sensitivity of averages to the relatively small number of samples collected. Towards the end of the simulation, the average stabilizes.

The queue delay graphs are compatible with the fact that for the same amount of traffic going into an interface, the outgoing link of higher bandwidth connected with the interface will allow that the interface queue be emptied faster. Therefore, the queue delays for faster links are smaller than those of slower links.

Note that the graphs above only show the delay in one direction, thus, the sum of the delays in both directions gives us the total end-to-end delay, which means that we only need to double the value of one direction in order to have the value for both directions.

11.1.4 NCN: Throughput - Links

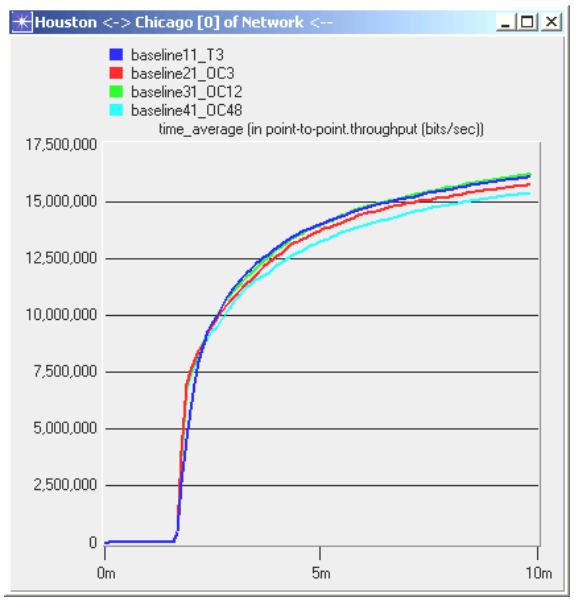


Figure 48: NCN: Throughput Hou-Chi

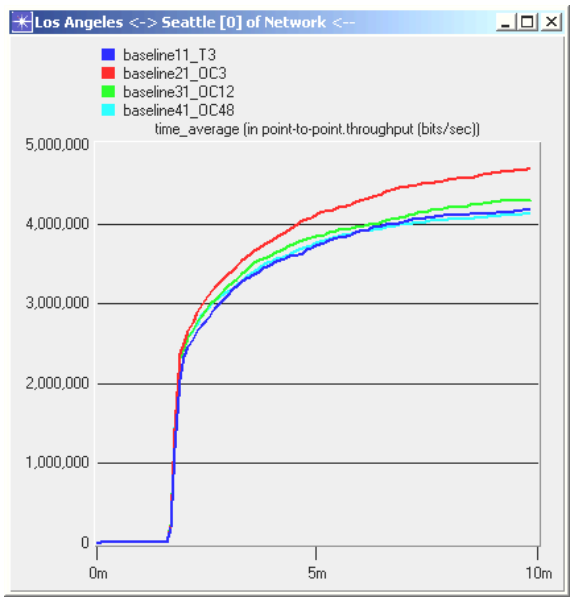


Figure 49: NCN: Throughput La-Seattle

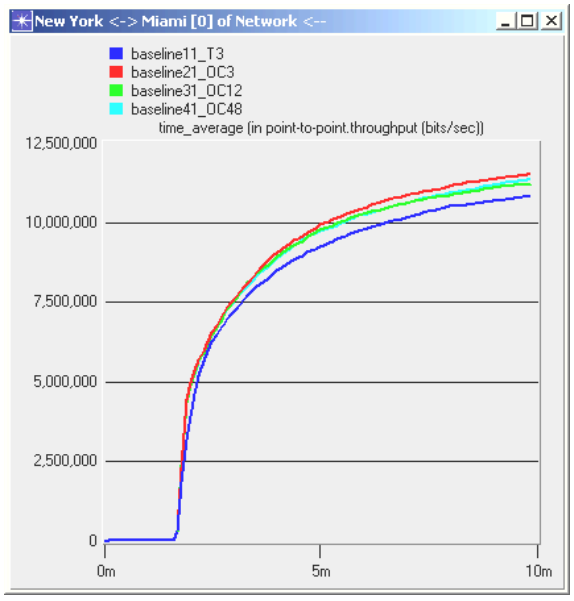


Figure 50: NCN: Throughput NY-Miami

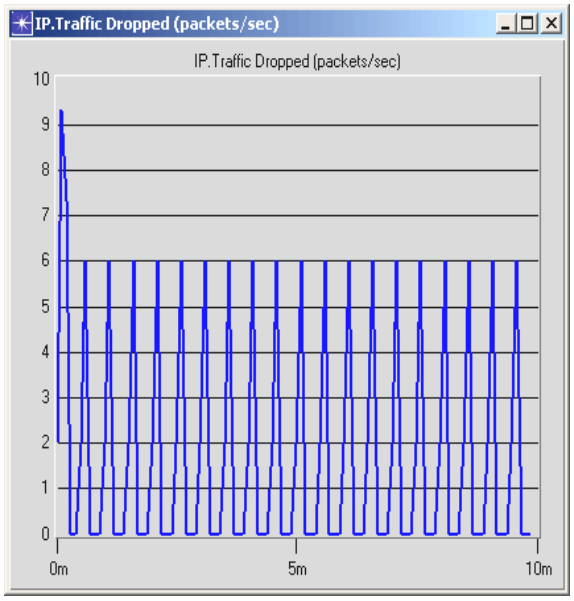


Figure 51: NCN: IP Traffic Dropped

The throughput graph for Houston-Chicago shows that towards the end of the simulation run the value stabilizes at around 15 Mbps, in one direction or 30 Mbps for both directions for each of the four links (T3, OC3, OC12, and OC48). With this amount of traffic the corresponding link utilization is: 67% for T3, 19% for OC3, 5% for OC12, and 1% for OC48. These utilizations are only for the link connecting Houston to Chicago. T3 is the best link to connect these two subnets, since it has the highest utilization with a value that still allows for network growth without risk of link saturation.

The throughput graph for LA-Seattle shows the same behavior as that of Houston-Chicago, i.e., at the end of the simulation run the value stabilizes at 8 Mbps for both directions for each of the links. And for this link the utilization is: 18% for T3, 5% for OC3, 1% for OC12, and 0.3% for OC48. T3 is again the best link to connect these two subnets, since it has the highest utilization with a value that still allows for network growth without risk of link saturation. The graph for NY-Miami reveals the throughput values converging around 22 Mbps for the duplex link. With this value of bandwidth, the utilization is 49% for T3, 14% for OC3, 4% for OC12, and 0.9% for OC48.

The graph for IP traffic dropped (not time-average but ‘as is’) shows an average of 2 packets being dropped during the simulation run. The reasons for dropping an IP datagram can be: insufficient space in the queue; maximum number of hops exceeded by an IP datagram; non routing nodes - a local router interface was not found to be used as the next, and routing node with the rote table lookup failed to yield a route to the destination. RIP (the routing algorithm) used by our routers, set the number of hops to 32.

11.1.5 NCN: Utilization - Links

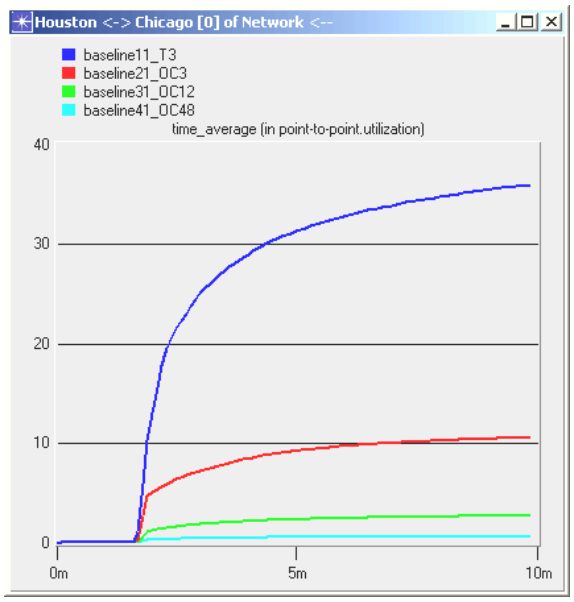


Figure 52: NCN: Utilization Hou-Chi (%)

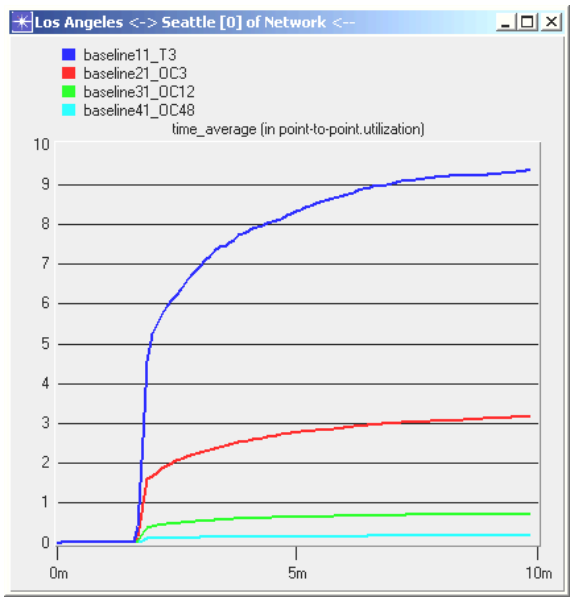


Figure 53: NCN: Utilization La-Seattle (%)

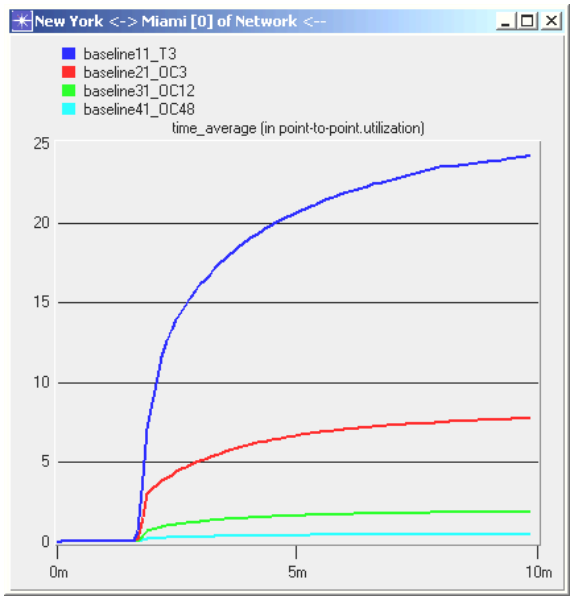


Figure 54: NCN: Utilization NY-Miami (%)

. Taking into account that either Houston or Chicago subnets are having traffic coming from three different subnets, the link utilization is higher than the ones for LA-Seattle and NY-Miami which each are interconnected with only two other subnets. For the load in the network, employing T3 as the link interconnecting the subnets represents better utilization of the network resources. The order of utilization is in reverse of the links bandwidths. Thus, OC-48 shows the lowest utilization for any of the pair of subnets.

. The utilization graphs are shown only for one direction of the packet flow, thus one needs to double every mean value mean in order to have the full link utilization. All the graphs are showing that the values of utilization are in line with the values of throughput that were shown before. Therefore, these two graphs are compatible and reflect a stable network configuration.. Table 14 shows the utilization for the subnet pairs and links.

	T-3	**OC-3**	**OC-12**	**OC-48**
Houston-Chicago	72	21	5	1
LA-Seattle	19	6	1	0.5
NY-Miami	48	16	4	1

Table 14: NCN Link Utilization (%)

Network designers abhors the extreme values of link utilization, that is, under-utilization or close to capacity utilization. Under these two situations, the network is taxed as a bad designed network, since the waste or the lack of resources is a sign of a bad design. Network managers can control and maintain the network reliability with a value between 10 to 70 percent of link utilization.

11.2 National Campus – Scenario 1F: T-3 link, Device Failure

In this configuration, one of the NY node distribution switches is set to fail. Taking into account the redundancy built in the model, we should not expect the interruption of the network traffic flow. This sensitivity test, which is commonly used by network designers, is one means to ascertain the proper setup of redundant network and network robustness under adverse conditions. The figure 55 shows the failed switch marked with an ”X” . The backup distribution switch should keep the network running and pass network traffic on to both switch-based LANs.

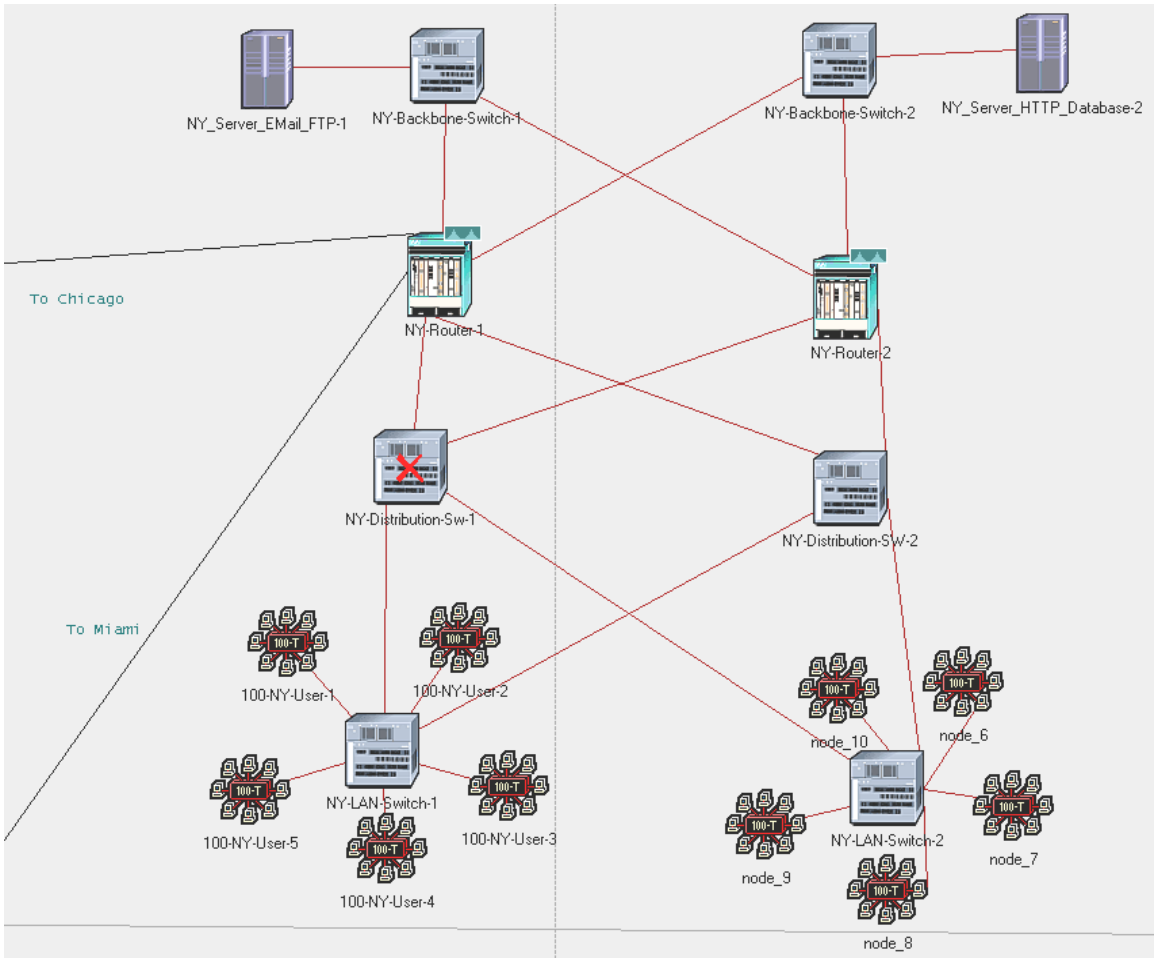


Figure 55: NY node - Failed Distribution Switch 1 (marked with an X)

11.2.1 NCN: ETE Delay – T-3 (Link Fail)

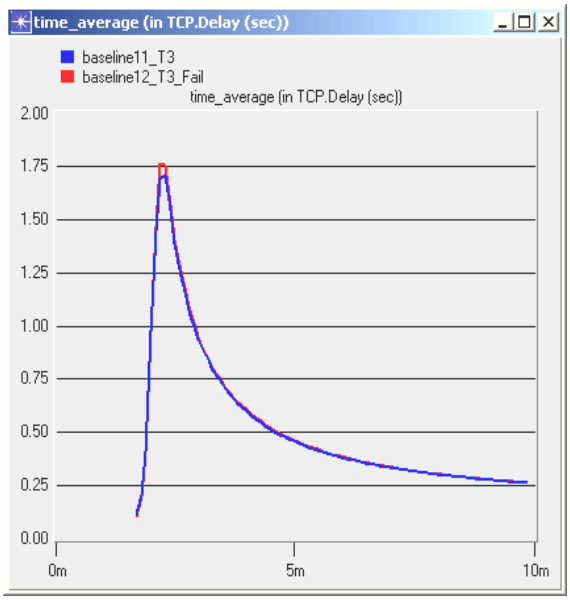


Figure 56: NCN: TCP delay

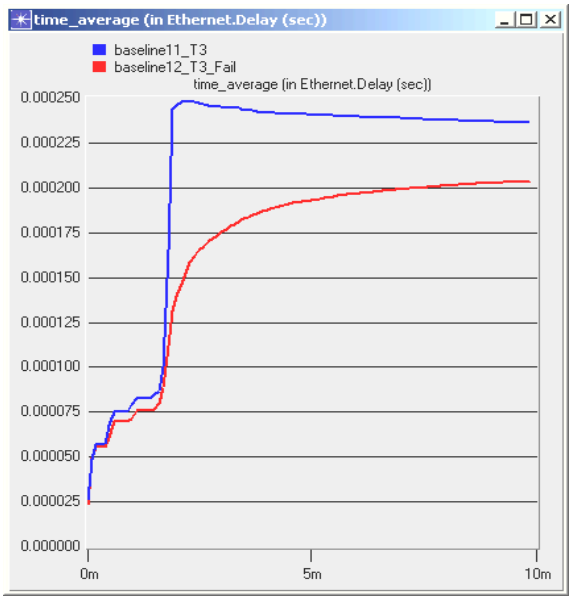


Figure 57: NCN: Ethernet delay (sec)

11.2.2 NCN: Response Time – T-3 (Link Fail)

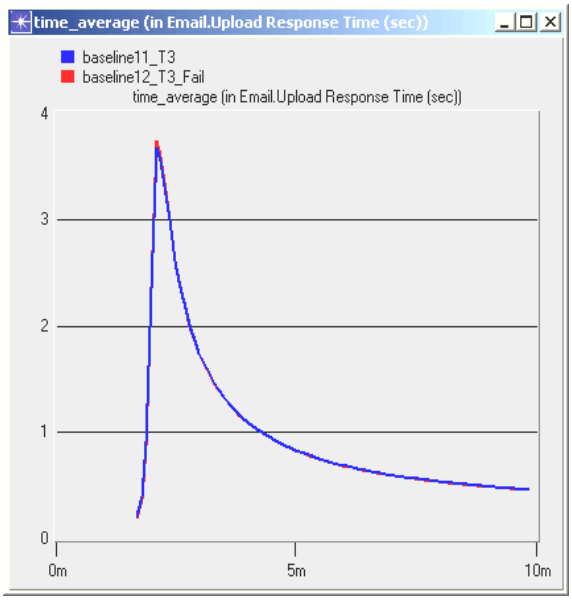


Figure 58: NCN: E-Mail response time

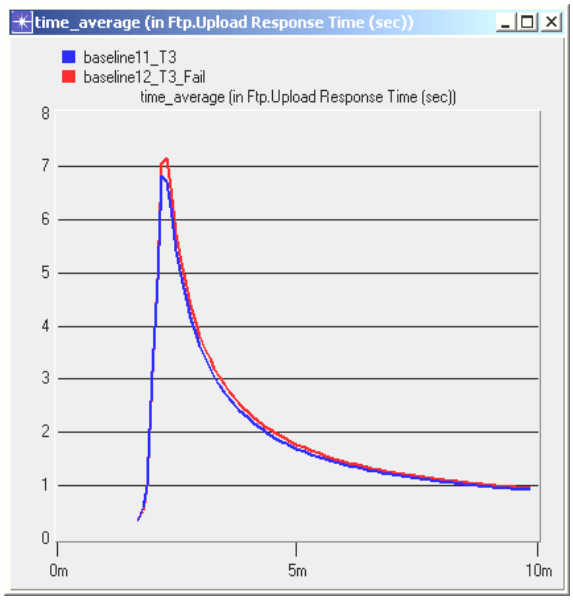


Figure 59: NCN: FTP response time

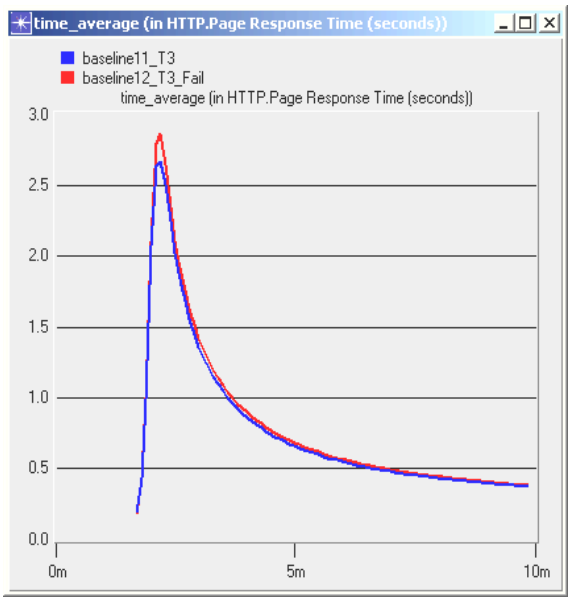


Figure 60: NCN: HTTP response time

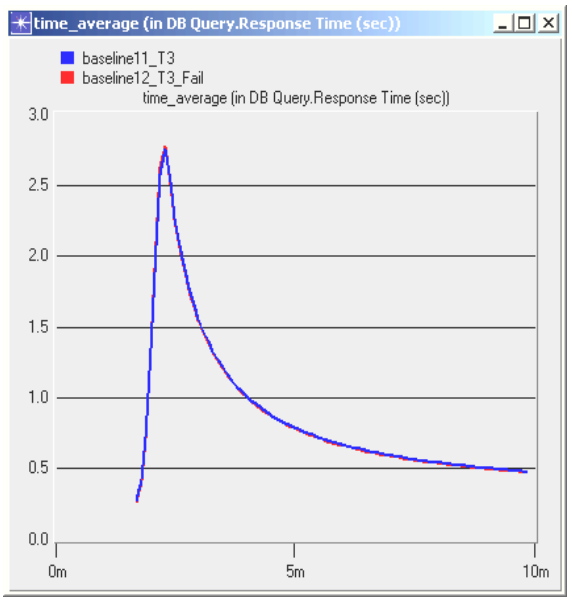


Figure 61: NCN: DB Query response time

11.2.3 NCN: Queue Delay – T-3 (Link Fail)

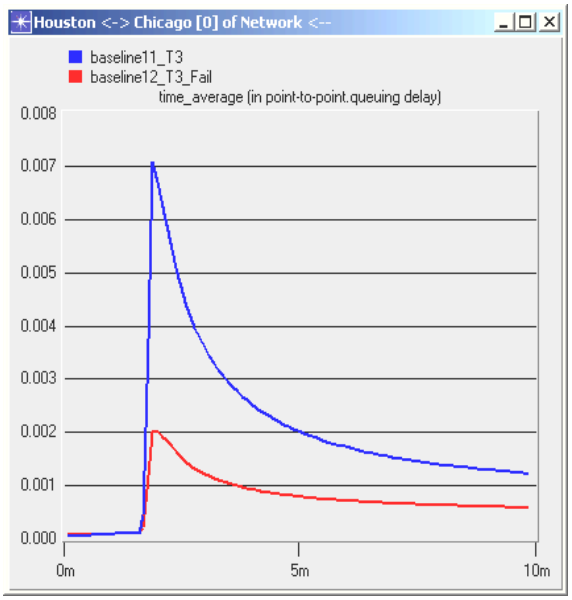


Figure 62: NCN: Queue delay Hou-Chi

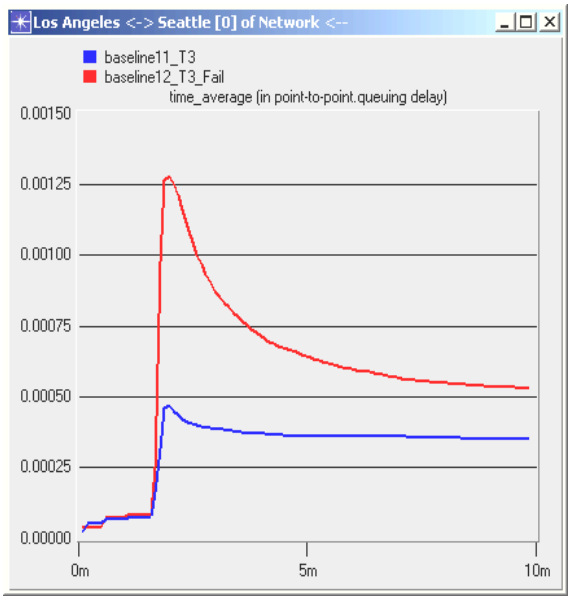


Figure 63: NCN: Queue delay La-Seattle

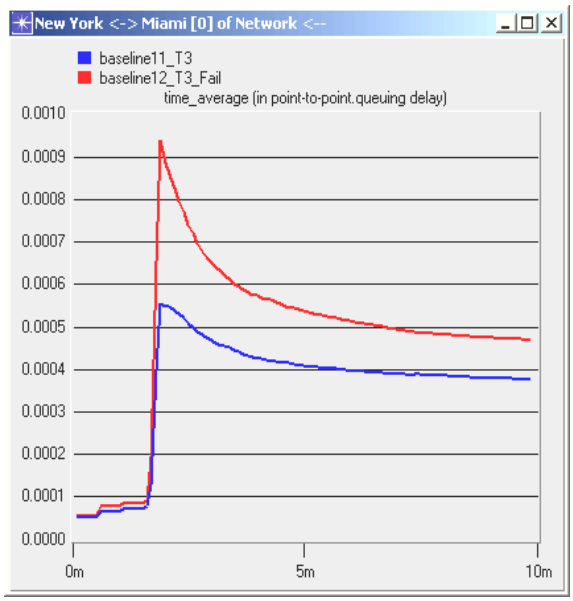


Figure 64: NCN: Queue delay NY-Miami

11.2.4 NCN: Throughput – T-3 (Link Fail)

Figures 69 and 70 show throughput for Houston-Chicago and LA-Seattle.

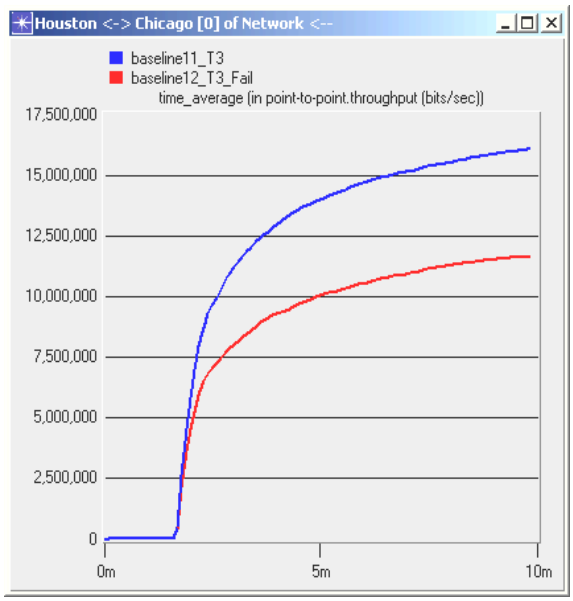


Figure 65: NCN: Throughput Hou-Chi

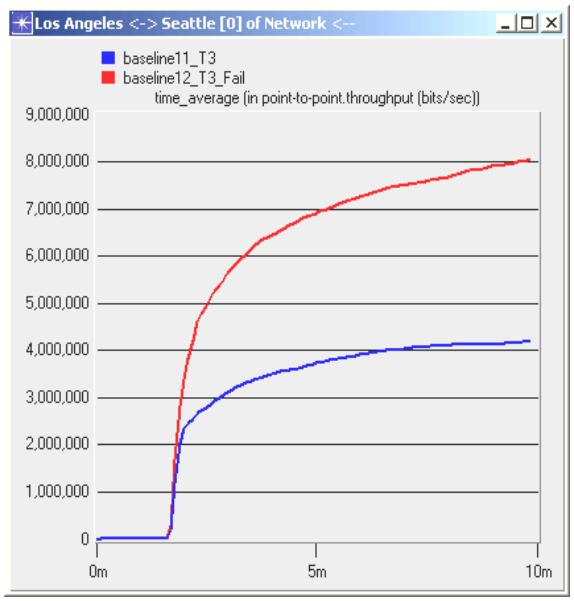


Figure 66: NCN: Throughput La-Seattle

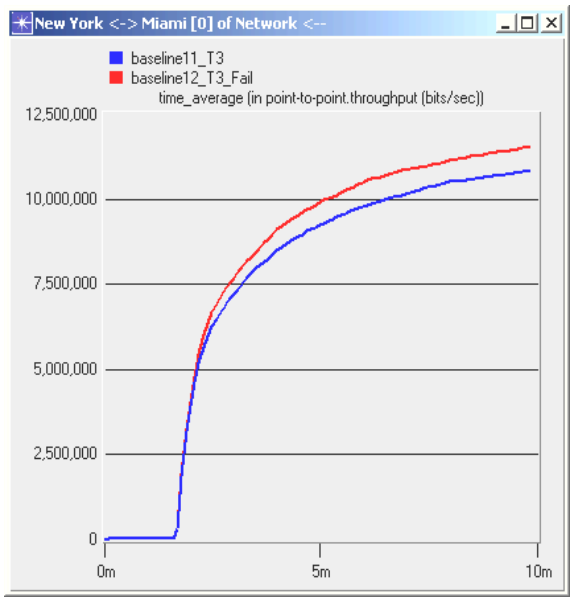


Figure 67: NCN: Throughput NY-Miami

11.2.5 NCN: Utilization – T-3 (Link Fail)

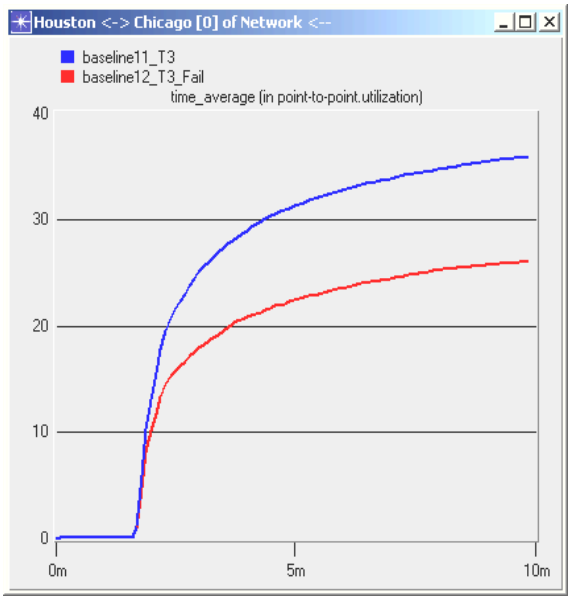


Figure 68: NCN: Utilization Hou-Chi (%)

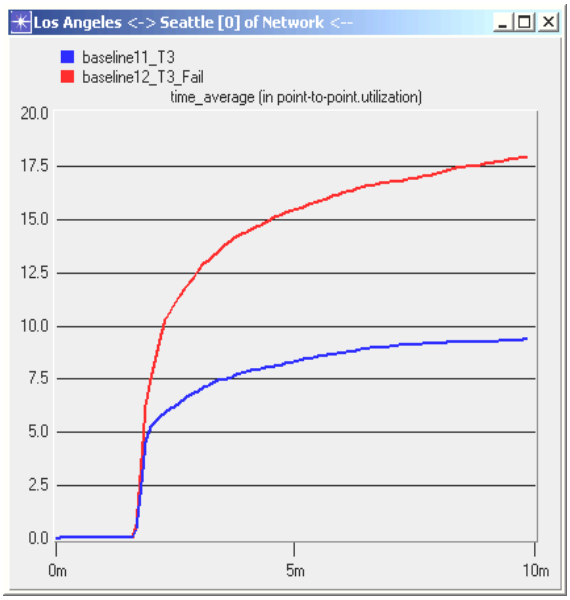


Figure 69: NCN: Utilization La-Seattle (%)

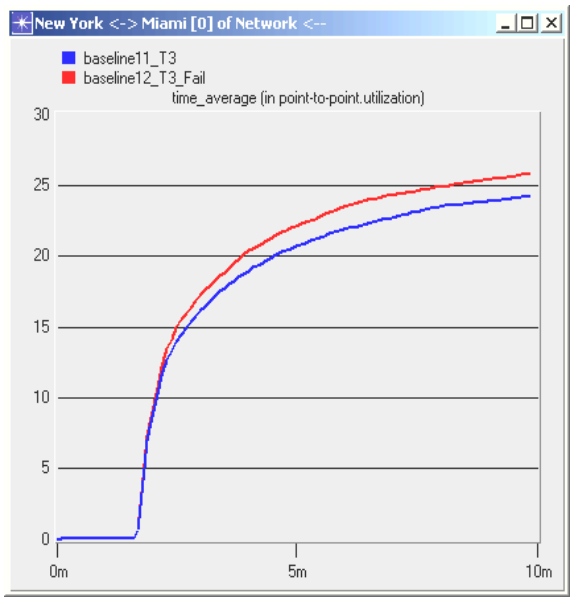


Figure 70: NCN: Utilization NY-Miami (%)

11.2.6 Analysis of the graphs

The main points that the above graphs show are:

. ETE delay: TCP delay shows that disabling the distribution switch 1 of a subnet did not affect the overall end-to-end delay of 0.25 sec. Ethernet delay presents a minor change (only 3 milliseconds slower) for the configuration with the disabled switch reflecting the fact that the communication between LAN and the router is a bit faster since a switch processing task is not necessary because one of the switches is not operational.

. Response Times: All the applications show no change whatsoever in their response times. Thus, it takes 0.6 seconds to get a response for e-mail, 0.9 seconds for an ftp, 0.4 seconds for a Web page, and 0.5 seconds for a database query.

. Queue delay: Packets are being held longer in the queues of NY-Miami link, because they are being sent to the router at a larger rate due to the fact that only one of the distribution switches is operational. The queue delay is now close to 1 millisecond, a increase of 25%. As a result, the queue delay for Houston-Chicago goes down by half to 1millisecond, and for LA-Seattle it goes up by fifty percent to 1 millisecond.

. Throughput: There is no change in the throughput for NY-Miami which is around 22 Mbps. The throughput for Houston-Chicago goes down by a quarter to 24 Mbps, and the throughput in the LA-Seattle link is doubled to 16 Mbps. This behavior reflects the effect that the queue delay at an interface has over the rate of data traffic going through the link connected to the interface. Notice that the queue delay for NY-Miami was up by a quarter which resulted in a corresponding decrease in the throughput of Houston-Chicago – any subnet is a link away of each other, while the subnets of LA and Seattle are four links away from NY and Miami subnets. LA-Seattle shows one hundred percent increase in the throughput. Also, note that Houston-Chicago is link that is at roughly mid-way point between the other two links being considered, LA-Seattle and NY-Miami.

. Utilization: The links utilization is directly related to the throughput, thus, the utilization of the Houston-Chicago link is down by a quarter percent to 52%, LA-Seattle is up 85 percent to 35%, and NY-Miami with no change and still using around 50% of the link.

. There were no reports of IP Traffic being dropped.

11.3 National Campus – Scenario 13: Link, Backload, and QoS

This scenario introduces two factors into the national campus network: background traffic and quality of service. The background traffic follows the previously established degradation levels, see table 9. The quality of service consists of setting up the routers with the ToS schema, which enables the routers to classify and order the incoming packets into the queue according to their ToS value. Thus, the routers are programmed to inspect the header of the IP datagram, look at the ToS value, and queue the datagram according to the ToS value (PHB value). The weighted fair queue (WFQ) schema is used to queue the datagrams, and it uses a RED method in case there is a need of discarding some packets due to congestion at the queue.

The queue size is increased to 1,000 packets to the database query applications, while all the other applications have a queue size of 500 packets. This is done in order to flush out any database packets that are waiting in the queue since the initial planning is to handle with higher priority these same packets. They were generated with a ToS(3), i.e. "Excellent Effort", which means they have priority over the other packets belonging to the other applications, and therefore they are handled expeditiously by the interfaces (routers) that are in charge of forwarding them. In summary, in the previous simulation runs the packets are enqueued in a single FIFO queue and no queue management algorithm is taking place. Now, a traffic conditioner is taking place for each packet and then the packets are enqueued in a WFQ queue with two sub-queues (FIFO and ToS) and RED is employed as the queue management algorithm. This mechanism for QoS is expected to result in better (shorter) response times and delays for our applications.

11.3.1 NCN: End-to-End Delay – Link + Backload + QoS

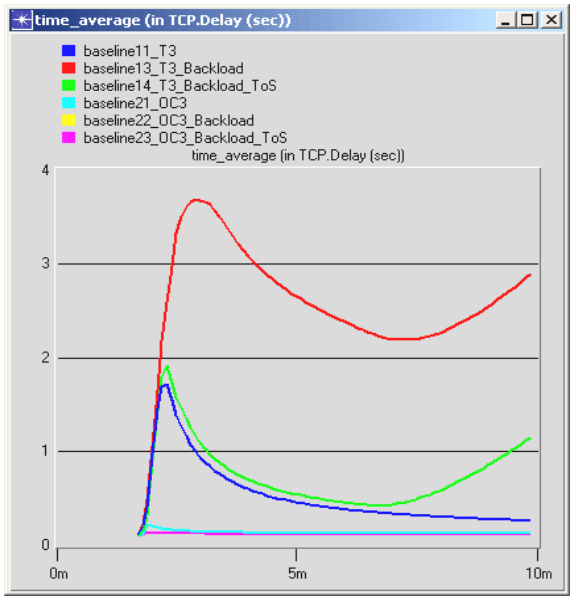


Figure 71: NCN: TCP delay T3/OC3

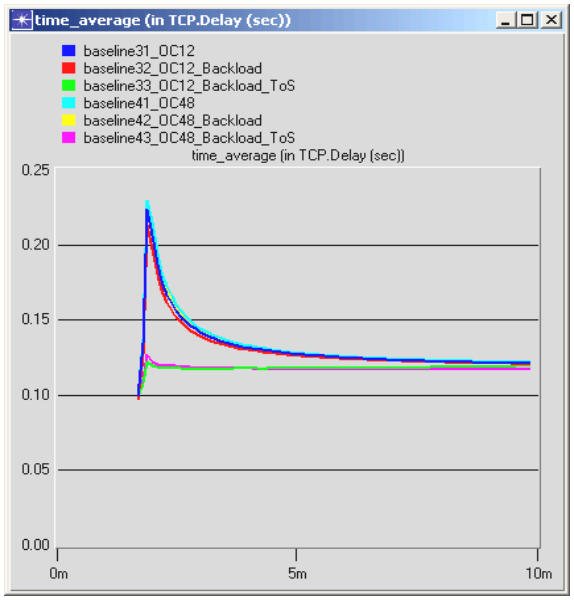


Figure 72: NCN: TCP delay OC12/OC48

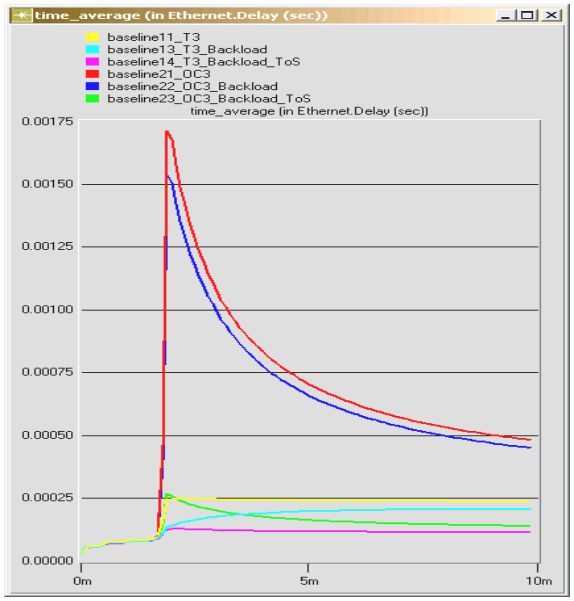


Figure 73: NCN: Ethernet delay T3/OC3

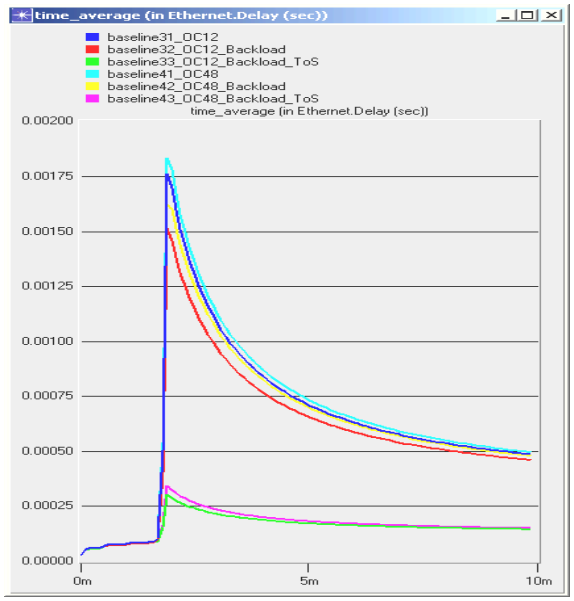


Figure 74: NCN: Ethernet delay OC12/OC48

Analysis of TCP delay: For the T3 link, the TCP delay for the link with backload and QoS is the same as the one with only T3 until the backload reaches 50 percent of the total traffic, and then at this point the delay moves upward ending at 1.15 seconds while the delay for the link with backload reaches a value of 2.9 seconds. This represents an improvement of the TCP delay because even though there is background traffic interfering with the explicit traffic of the applications, the QoS schema via ToS shows that at lower level of degradation (30 to 40 percent) the overall TCP delay is unaffected by the backload traffic.

For the optical cables (OC3, OC12, and OC48), the improvement is still present at lower levels of degradation, however when it reaches 50% of the explicit traffic, the TCP delay is practically not affected by the degradation level and QoS, settling for a constant value of 0.125 seconds.

Regarding the ETE Ethernet delay, the T3 link distributions show that the Ethernet delay for the link with a backload traffic was reduced from 0.23 to 0.20 ms, and with the introduction of QoS, the delay was reduced even further to 0.12 ms. Again, an improvement in the network performance when QoS is used.

The Ethernet delay is reduced from 0.5 ms for the optical cables with or without the backload traffic to 0.2 ms for the cables with backload and QoS; thus reflecting the capacity of the QoS schema in reducing the amount of delay to more than half.

11.3.2 NCN: Response Time – Link + Backload + QoS

- E-Mail

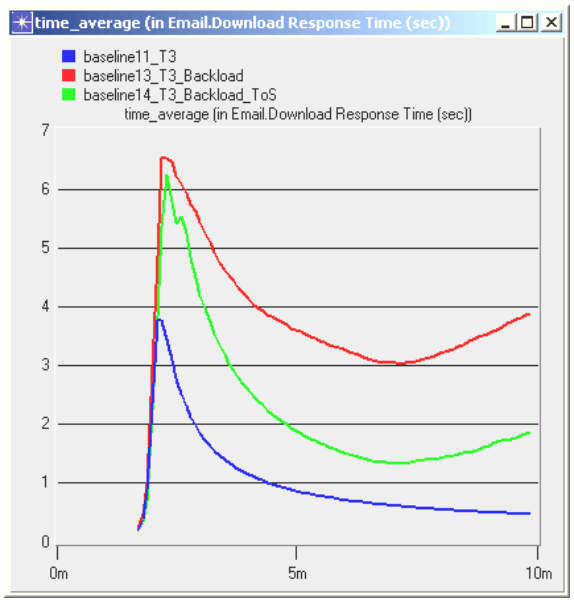


Figure 75: NCN: E-Mail response time T3

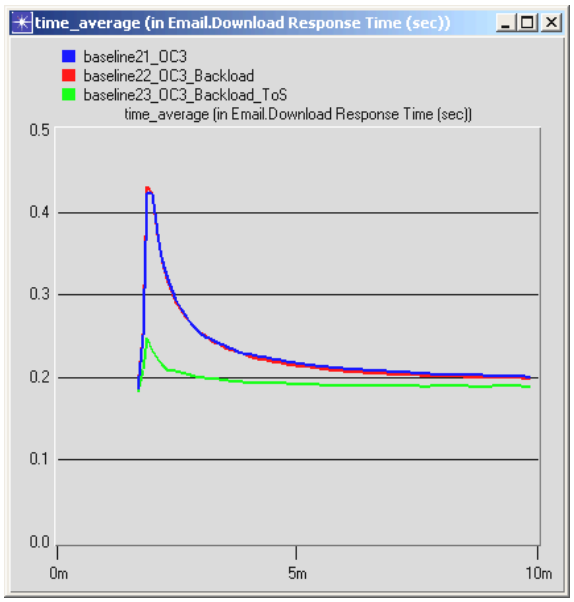


Figure 76: NCN: E-Mail response time OC3

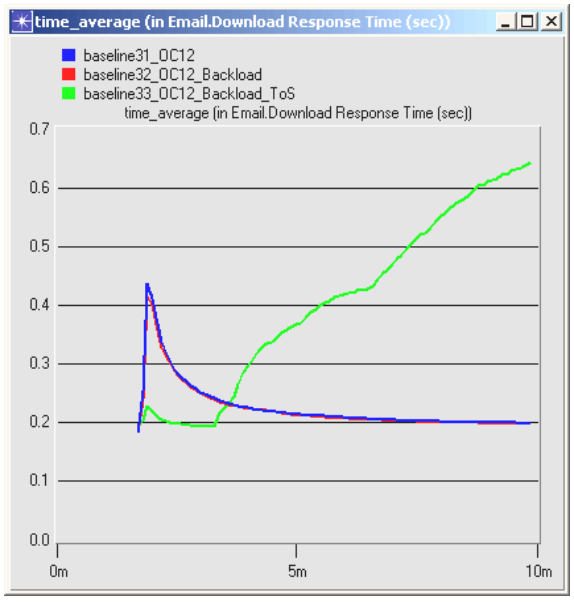


Figure 77: NCN: E-Mail response time OC12

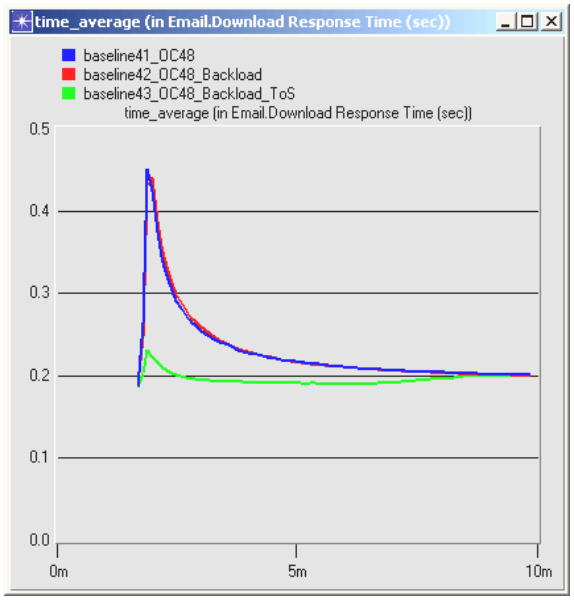


Figure 78: NCN: E-Mail response time OC48

- FTP

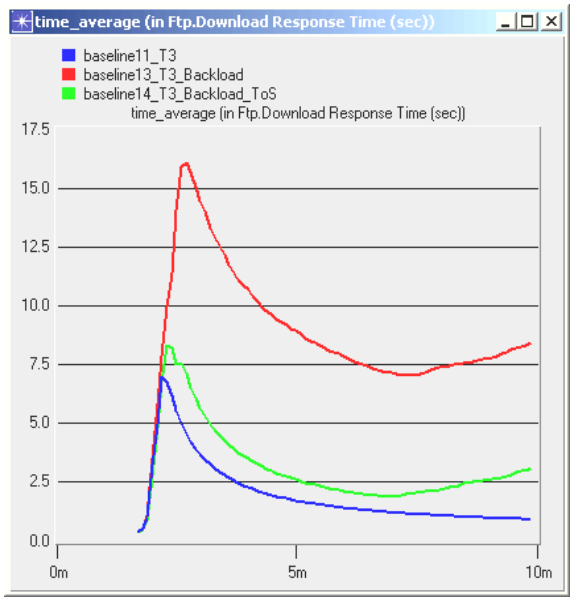


Figure 79: NCN: FTP response time T3

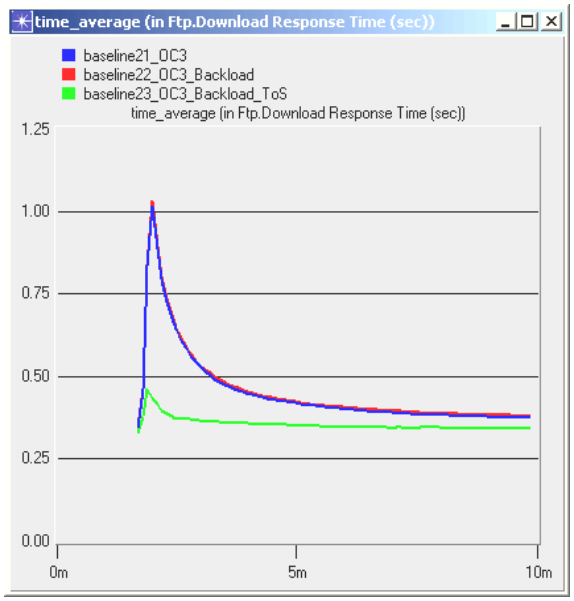


Figure 80: NCN: FTP response time OC3

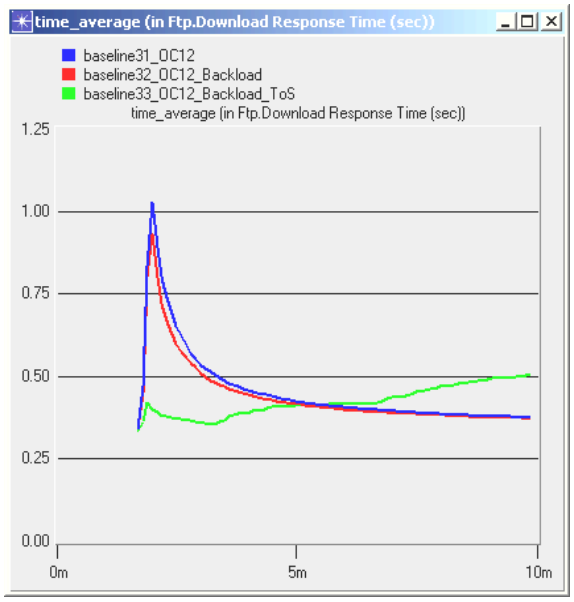


Figure 81: NCN: FTP response time OC12

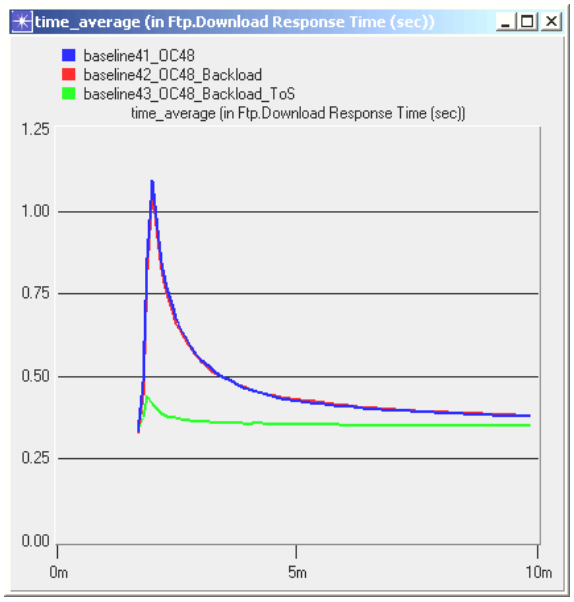


Figure 82: NCN: FTP response time OC48

- HTTP Browsing

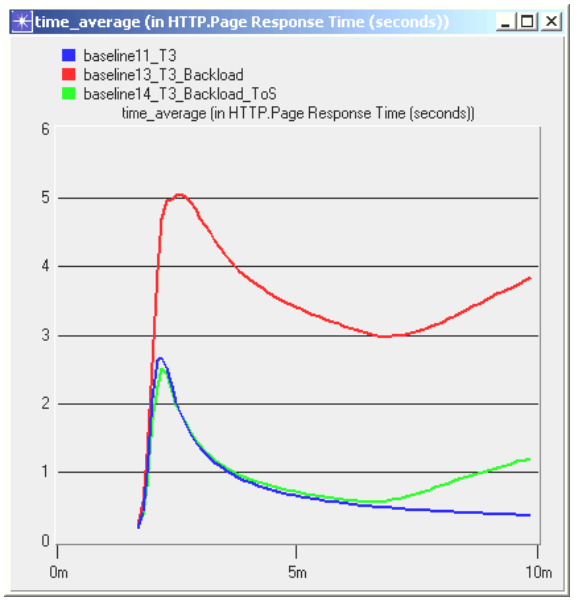


Figure 83: NCN: HTTP response time T3

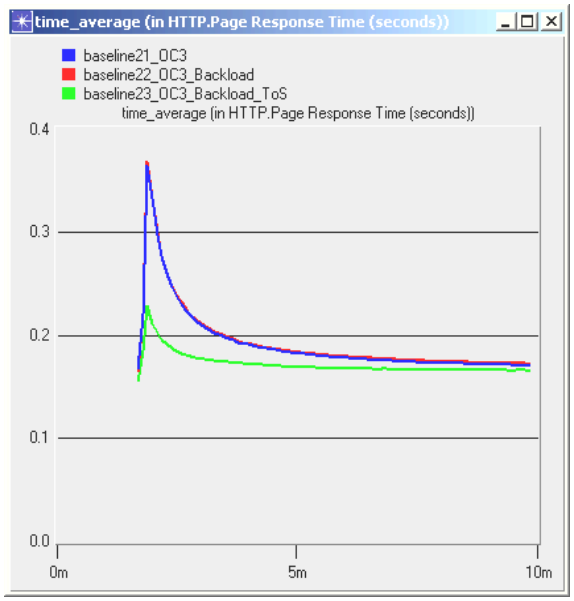


Figure 84: NCN: HTTP response time OC3

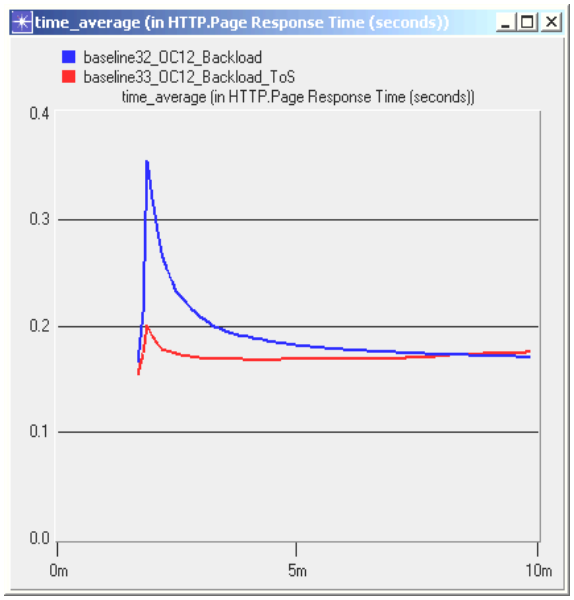


Figure 85: NCN: HTTP response time OC12

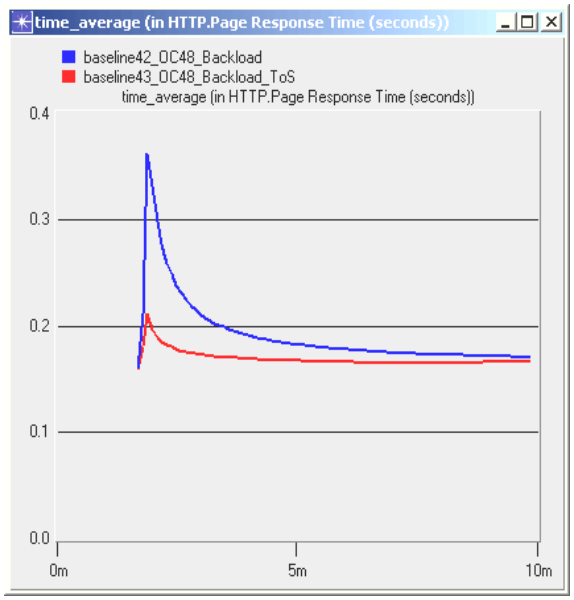


Figure 86: NCN: HTTP response time OC48

- Database Query

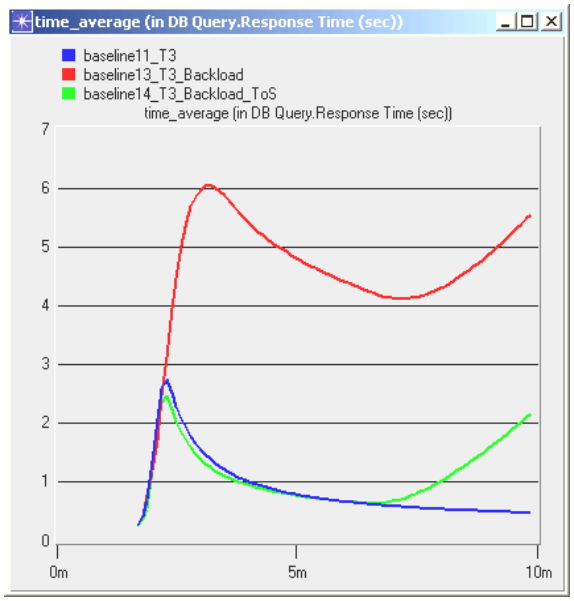


Figure 87: NCN: DB Query T3

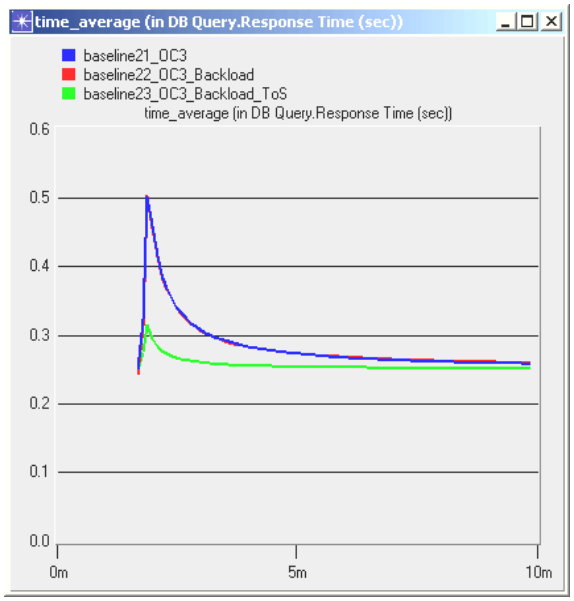


Figure 88: NCN: DB Query OC3

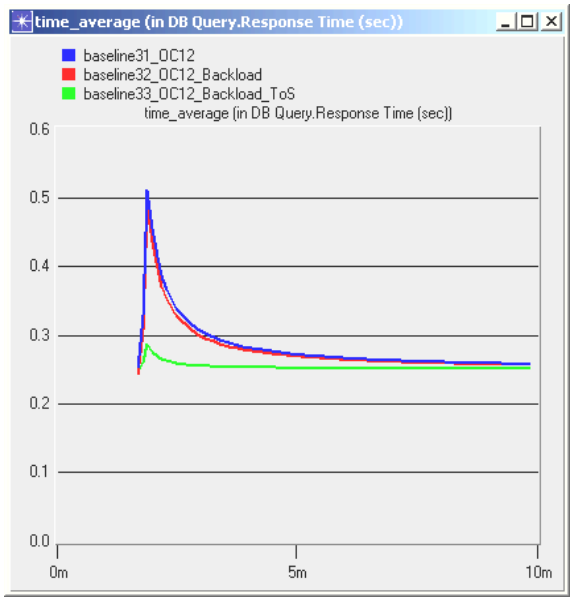


Figure 89: NCN: DB Query OC12

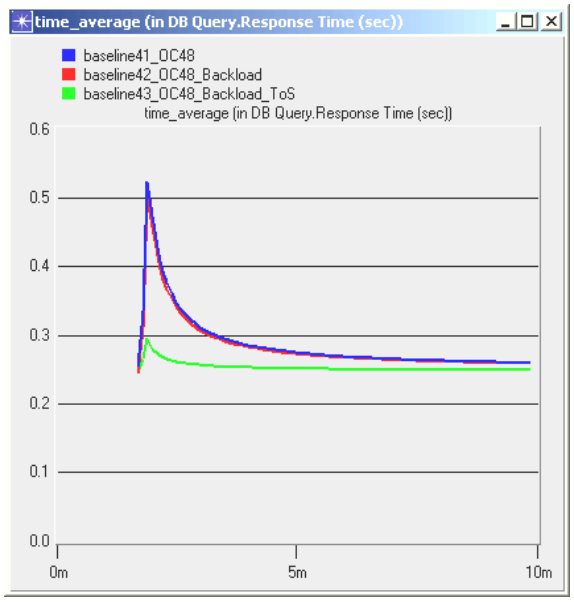


Figure 90: NCN: DB Query OC48

E-Mail: The T3 distributions show that when background traffic is injected into the network, the response time for emails goes up to 4 seconds, and when QoS is added to the traffic then the response time goes down to 2 seconds, which represents 100% decrease in the response time. The email response time for undisturbed T3 link is 0.5 seconds. One striking feature is that while the background traffic increases from 30% to 40% the response time decreases but it increases when the background becomes 50% of the traffic. For the optical cables, the background traffic does not affect the response time, reaching a constant value of 0.2 seconds. For lower level of degradation, 30% of traffic, QoS reduces the response time by 0.05 seconds, however as the degradation increases to 40 or 50% of the traffic then the QoS either does not affect the response time for OC3 and OC48, or it shows a remarkable increase in the response time for OC12, jumping from a low 0.2 seconds to a high 0.6 seconds at the end of the simulation run.

FTP: The T3 distributions show that when background traffic is injected into the network, the response time for ftp goes up to 7.9 seconds, and when QoS is added to the traffic then the response time goes down to 2.6 seconds, which represents 67% decrease in the response time. The email response time for undisturbed T3 link is 1.2 seconds. The same striking feature happens again, while the background traffic increases from 30% to 40% the response time decreases but it increases when the background becomes 50% of the traffic. For the optical cables, the background traffic does not affect the response time, reaching a constant value of 0.38 seconds. For lower level of degradation, 30% of traffic, QoS reduces the response time by 0.14 seconds, however as the degradation increases to 40 or 50% of the traffic then the QoS either does not affect the response time for OC3

and OC48, or as with the email response time it shows a remarkable increase in the response time for OC12 around the time 50% of the traffic corresponds to background traffic, jumping from a low 0.38 seconds to a high 0.50 seconds at the end of the simulation run.

HTTP: The T3 graphs reveals that there is a 3 seconds jump in the response time for HTTP when a background traffic is present in the network. One can observe the same behavior of the HTTP response time with the different levels of degradation: from 30 to 40% backload the response time decreases, and it increases when the backload becomes 50% of the traffic. This reflects the sensitivity of the application with regards to the backload traffic. With QoS, the response time shows is the same as that of the undisturbed link of 0.6 seconds until the backload becomes 50% of the traffic, then the response time increases to 1.2 seconds while the undisturbed link settles for 0.5 seconds.

Analysis of the optical cables distributions shows the same pattern as before: the QoS schema reduces the response time by 0.03 seconds while the backload is only 30% of the traffic. Over this level of degradation, QoS does not affect the response time which becomes a constant value of 0.17 seconds.

DB Query: The same behavior as seen for the other applications is shown for DB Query. With the T3 link, QoS neutralizes the negative effect of the background traffic over the response time making it equal to the undisturbed link until the background becomes 50% of the traffic. Then the response time increases from 0.6 seconds to 2.18 seconds while

the undisturbed links ends at a constant 0.5 seconds. While the backload shows the same behavior, a decreasing of the response time to a low 4 seconds as the background goes from 30 to 40% of the traffic, and then the response time goes up to 5.6 seconds during the time that the background is risen to 50% of the traffic.

Regarding the optical cables, not much of a change, QoS reduces the response time by the same 0.03 seconds during the 30% background traffic, but it does not affect the response time for higher values of the background traffic. The response is a constant 0.26 seconds. Also, the background traffic does not affect the response time for database queries.

11.3.3 NCN: Queue Delay – Link + Backload + QoS

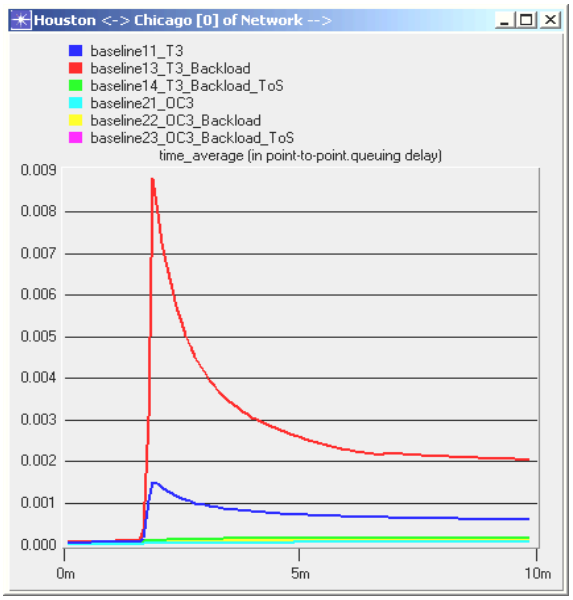


Figure 91: NCN: Q delay Hou-Chi T3/OC3

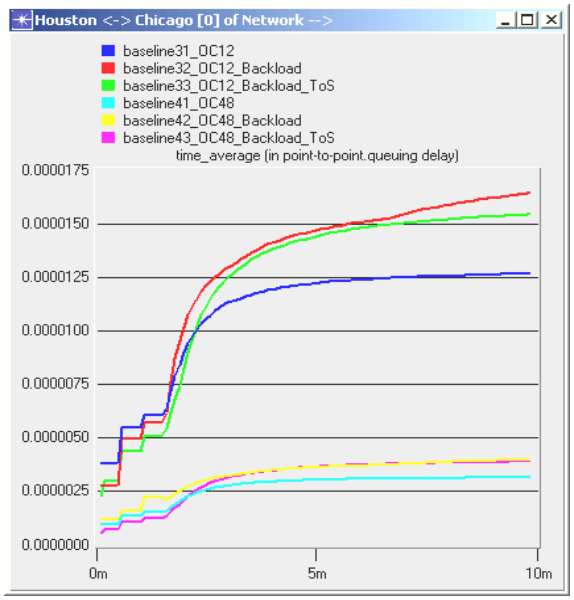


Figure 92: NCN: Q delay Hou-Chi OC12/OC48

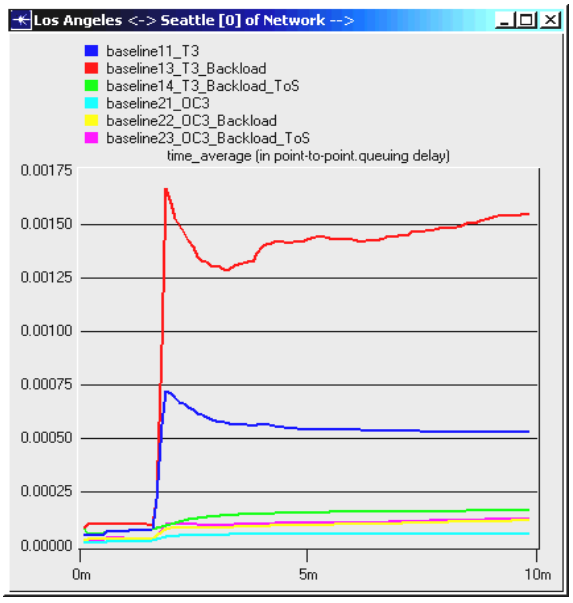


Figure 93: NCN: Q delay LA-Sea T3/OC3

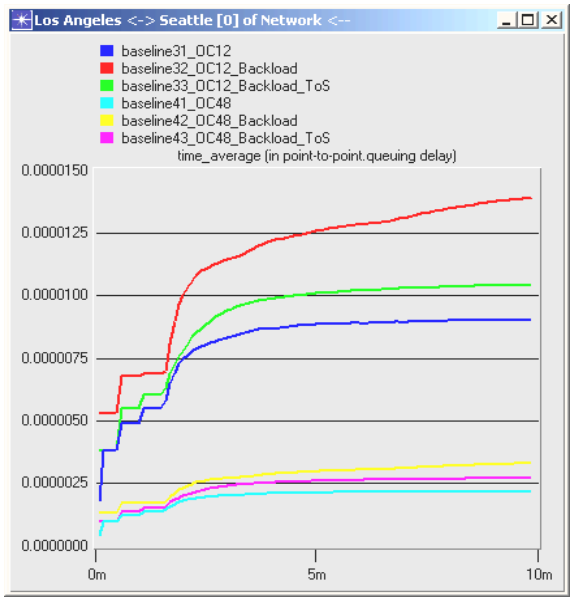


Figure 94: NCN: Q delay LA-Sea OC12/OC48

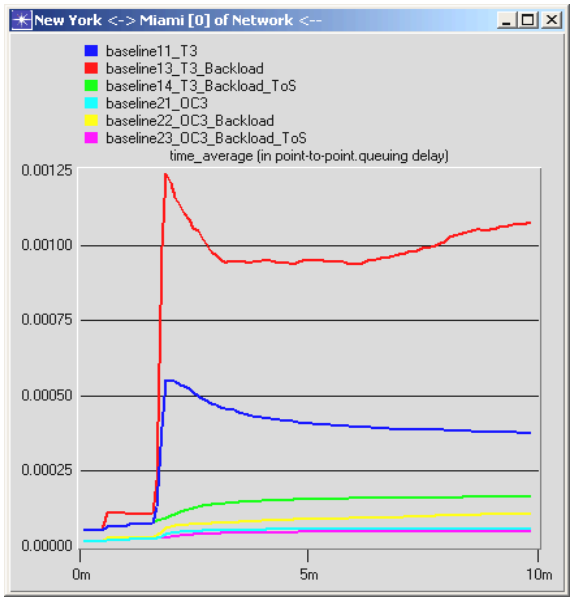


Figure 95: NCN: Q delay NY-Mia T3/OC3

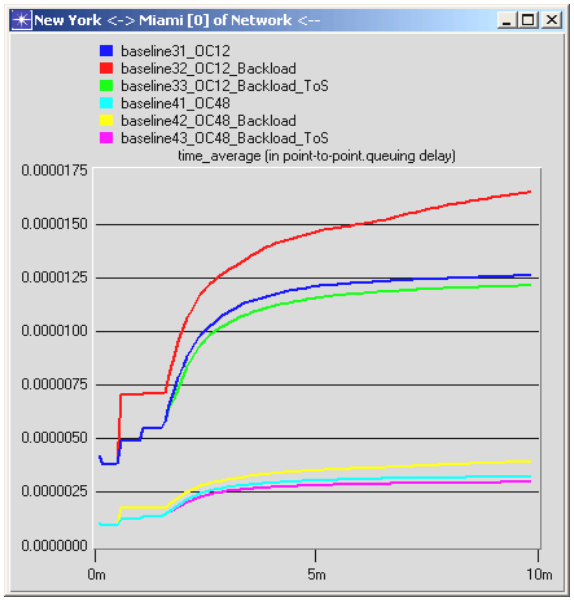


Figure 96: NCN: Q delay NY-Mia OC12/OC48

The Queue delay graphs for the chosen pair of subnets (Houston-Chicago, LA-Seattle, and NY-Miami) show common features:

. T3: The delay reaches a maximum of 2 milliseconds for the T3 link between Houston-Chicago when there is background traffic, and under 1 millisecond for the undisturbed link. However, when QoS is present and T3 is the link, the delay goes down about 50%, which it shows that the packets stay in the queue half of the time than that when QoS is not used.

. OC3/OC12/OC48: The delays are quite small for the undisturbed link because the traffic generate by the applications are quite low for the high bandwidth of the optical cables.

Although with the background traffic the throughput increases considerably, but the queue delay is still below a maximum of 0.0175 milliseconds. But one can still see that with QoS present the delay is reduced by close to 50% of that of the link with background traffic.

Our three scenarios consist of the first scenario with the undisturbed link carrying the applications traffic (explicit traffic), the second scenario with a hybrid traffic of explicit traffic and background traffic, and finally our third scenario running the hybrid traffic but now quality of service is present in the network.

Therefore, when the graphs show a reduction in the delay with QoS-enabled traffic, then it means that the QoS prevails over the affect of the background traffic. And that's what it is happening now.

11.3.4 NCN: Throughput – Link + Backload + QoS

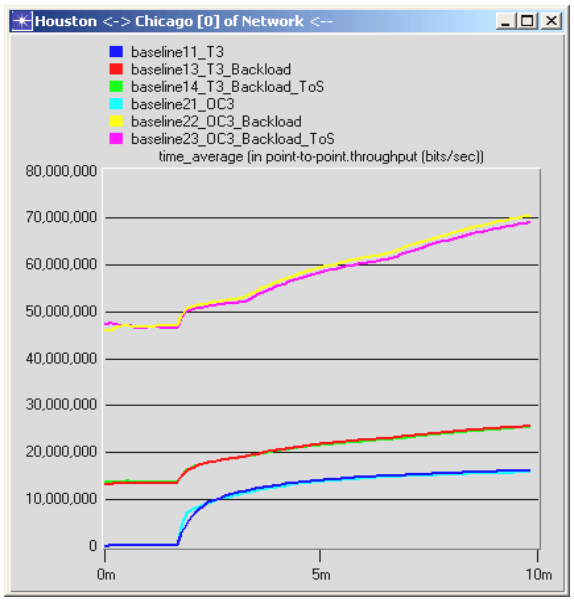


Figure 97: NCN: Thruput Hou-Chi T3/OC3

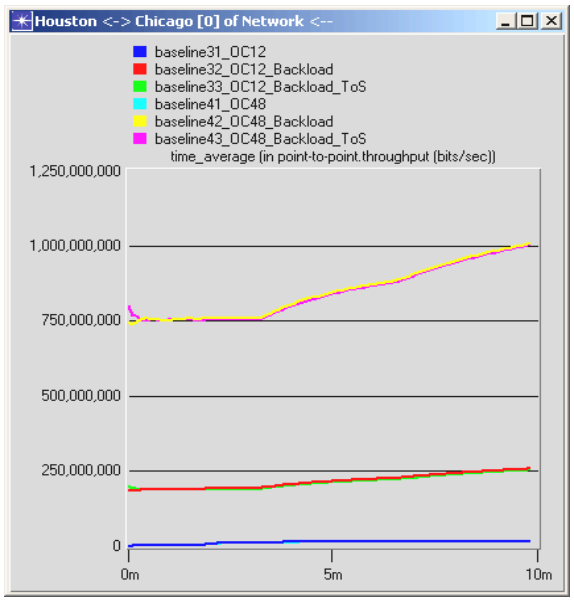


Figure 98: NCN: Thruput Hou-Chi OC12/OC48

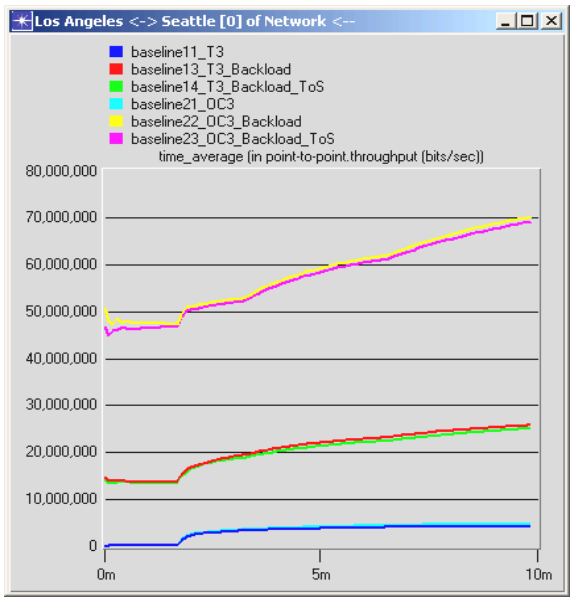


Figure 99: NCN: Thruput LA-Sea T3/OC3

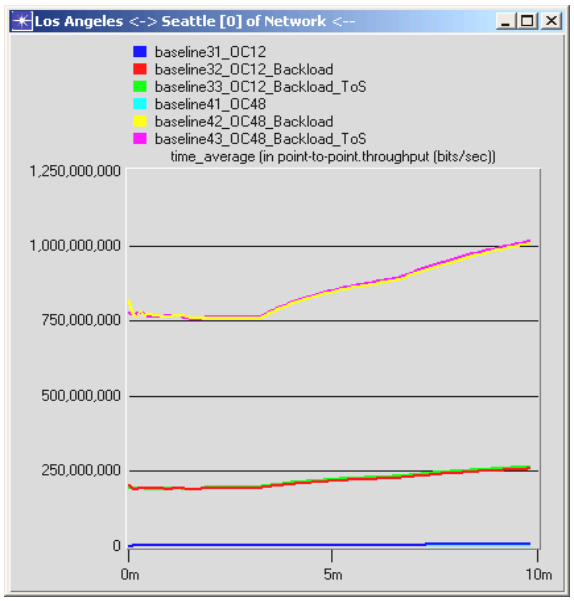


Figure 100: NCN: Thruput LA-Sea OC12/OC48

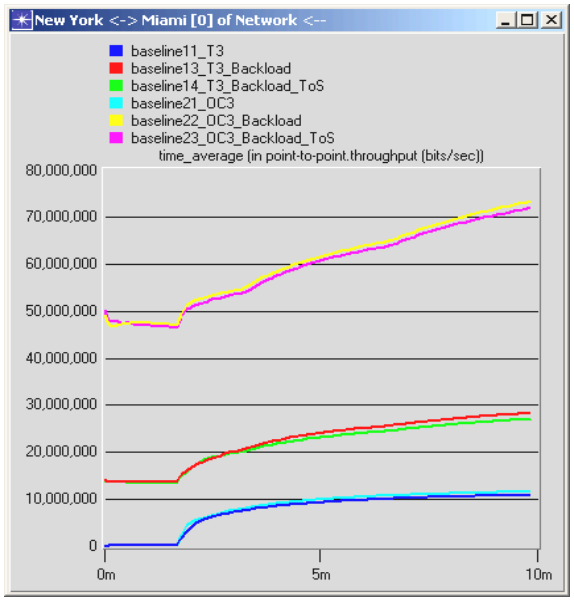


Figure 101: NCN: Thruput NY-Mia T3/OC3

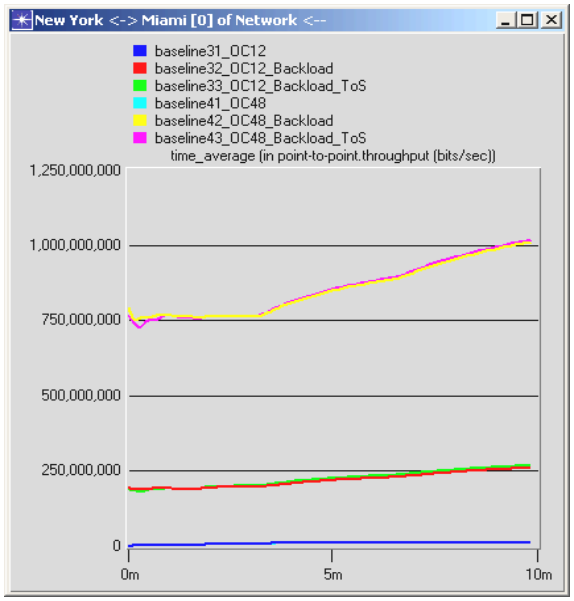


Figure 102: NCN: Thruput NY-Mia OC12/OC48

. Throughput is not influenced by QoS, and that is shown in all of the above graphs.

. The mean throughput for the undisturbed link is the same for all of the different cables and it assumes the following mean values for the different pair of subnets: 22 Mbps for Houston-Chicago, 6.2 Mbps for LA-Seattle, and 15.4 Mbps for NY-Miami. The high traffic Houston-Chicago link affects the traffic for the NY-Miami link since each of subnets is directly connected to either Houston or Chicago.

. The mean throughput for the hybrid traffic (explicit and background) for T3 link is 42.2 Mbps, 118 Mbps for OC3, 434.8 Mbps for OC12, and 1,706.6 for OC48.

11.3.5 NCN: Utilization – Link + Backload + QoS

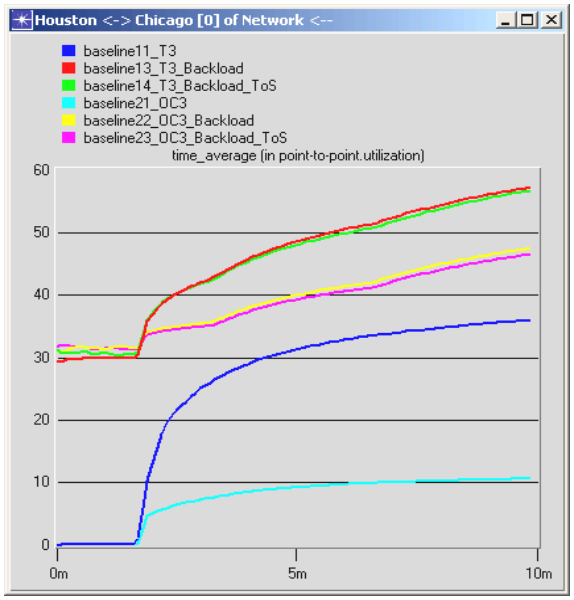


Figure 103: NCN: Utilization Hou-Chi T3/OC3

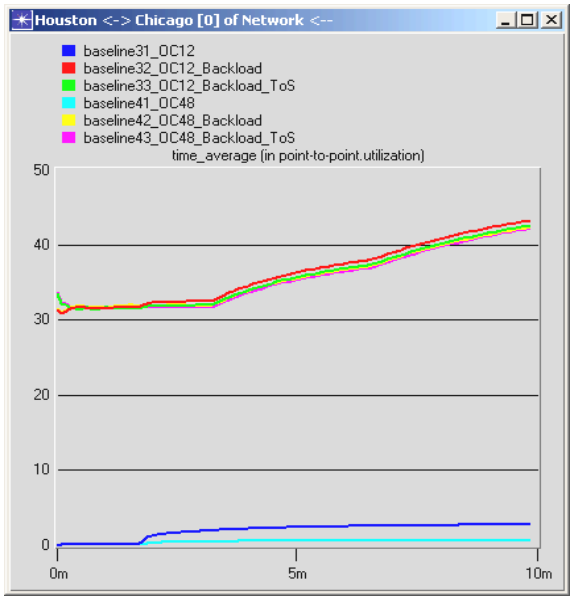


Figure 104: NCN: Utilization Hou-Chi OC12/OC48

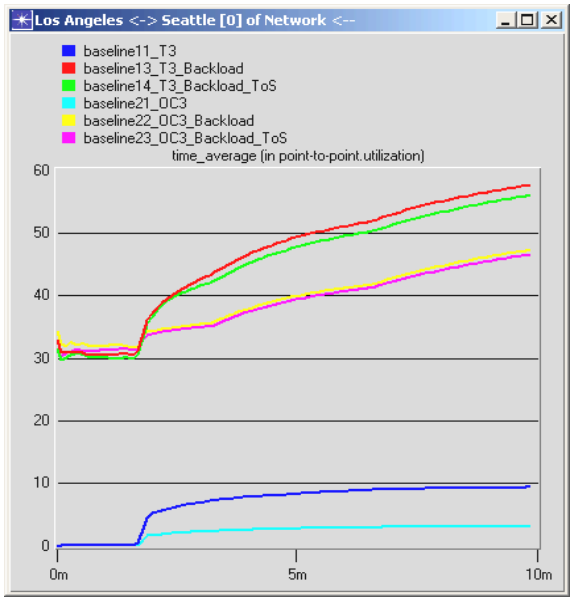


Figure 105: NCN: Utilization LA-Sea T3/OC3

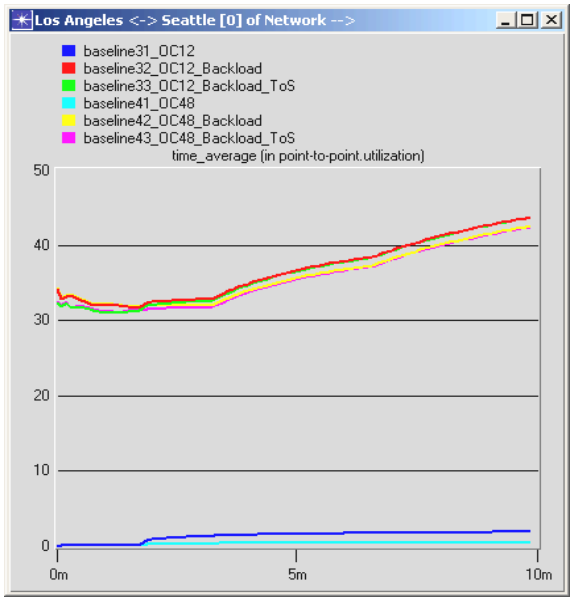


Figure 106: NCN: Utilization LA-Sea OC12/OC48

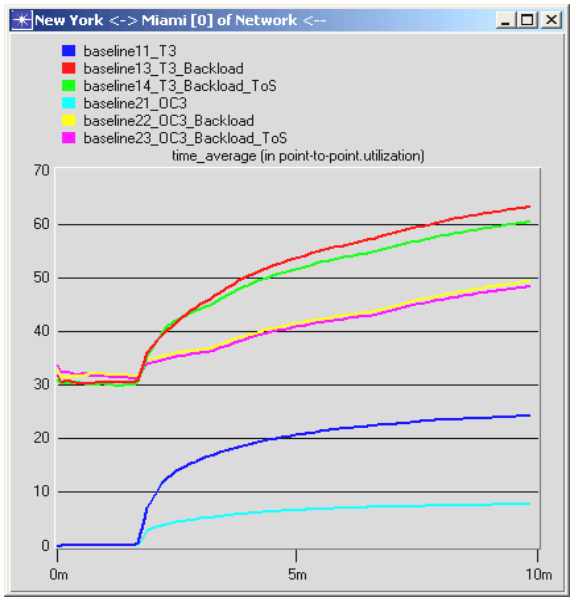


Figure 107: NCN: Utilization NY-Mia T3/OC3

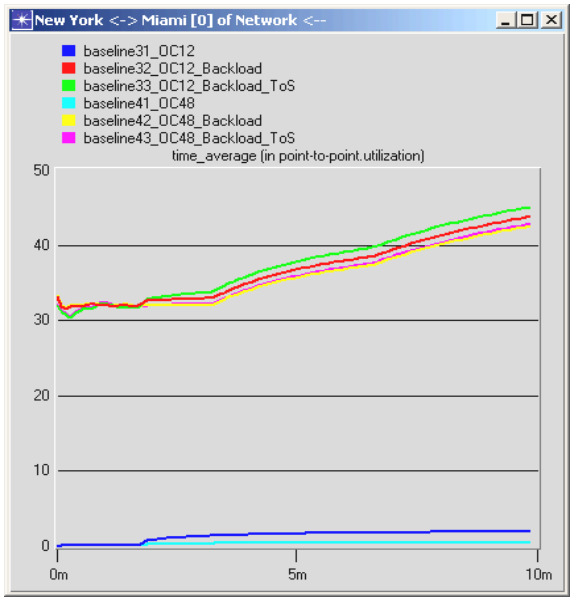


Figure 108: NCN: Utilization NY-Mia OC12/OC48

. Utilization is not affected by QoS.

. As it is to be expected, link utilization is a function of the throughput and the link bandwidth, being that the throughput is constant for over the link connecting the subnets pairs in consideration, therefore we have different utilizations for each cable being used (T3, OC3, OC12, OC48) for each pair of subnets: Houston-Chicago (50, 14, 4, 1), LA-Seattle (14, 4, 1, 0.2), and NY-Miami (34, 10, 2, 0.6).

. The link utilization for hybrid traffic (explicit plus background) is independent of the pair of subnets in study (Houston-Chicago, LA-Seattle, and NY-Miami), only differs for the different type of link: 95% for T3, 80% for OC3, 73% for OC12, and 72% for OC48.

11.4 International Campus – Scenario 21 – Links

The international campus network (ICN) is configured as an extension of the national campus network (NCN), with the original six subnets placed around the USA plus five other subnets, three of them located in Europe (London, Paris, and Berlin), one in Japan, and one in Brasil. All the subnets are interconnected via full-duplex links through the core subnet routers. Each simulation runs for a different link (T-3, OC-3, OC-12, and OC-48) for 10 minutes. The connection between workstations and switch as well as the connection between switch and the servers are 100Base-T full-duplex links. A gigabit Ethernet link connects the switches to the routers. As with all the configurations, the user profile is our "Heavy User Ethernet Client". The metrics, global and object statistics, are also the same as with all the previous scenarios, that is, end-to-end delay (TCP and Ethernet), response time (DB Query, HTTP, FTP, and E-Mail), and link (queue delay, throughput, and utilization). ICN link object statistics are the same links as the ones we selected for the NCN plus the connections to the international nodes.

These are the selected links:

Houston – Chicago	New York – Miami	Los Angeles – Seattle
New York – London	Miami – Sao Paulo	Los Angeles – Tokyo

One notable feature in some of the graphs is the fact that the distribution for ICN configurations in which T-3 is the link that interconnects the subnets skews the rest of the distributions, since with T-3 link the load results in extremely high link utilization (over 100%) and consequently extremely high queue delays. Therefore, one needs to have

some of the graphs without T-3 being included in order to better gauge the performance of the network for the optical fiber links.

Before showing the simulation results for different scenarios, figure 109 is the comparison of the international campus network traffic generated by e-mail, ftp, db query, and http.

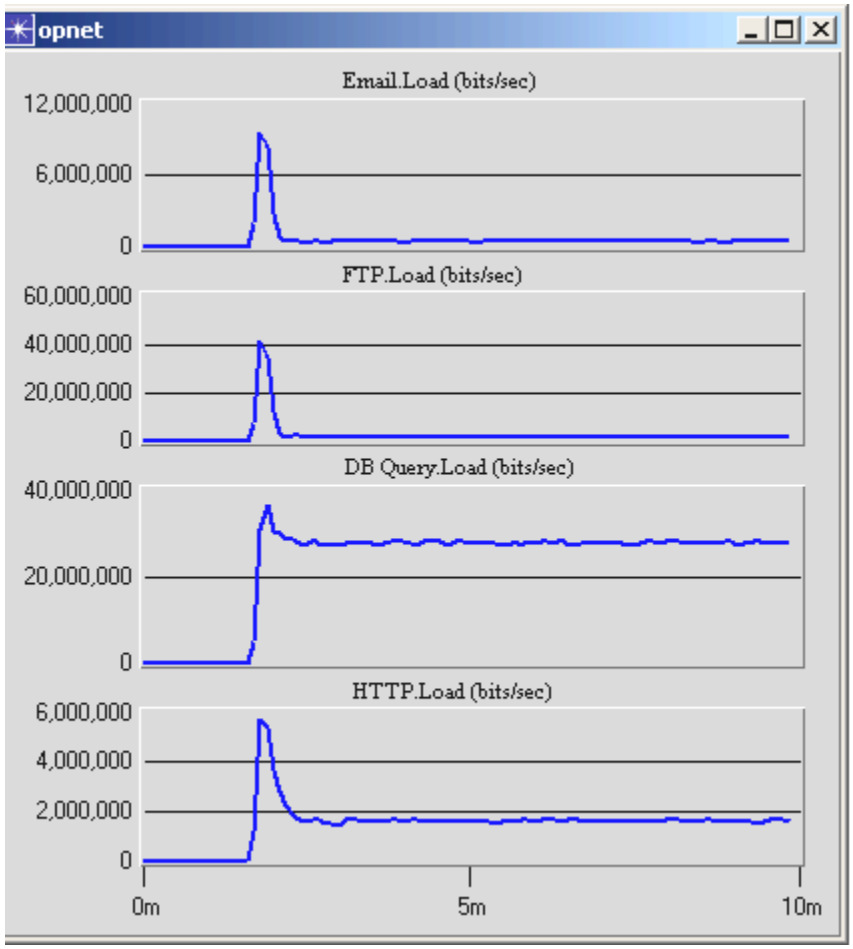


Figure 109: ICN: Traffic load comparison

The peaks we see in figure 109 indicate the time, between 100 and 110 seconds, when the data traffic starts to flow through the links of the network. After the peak the traffic load is almost constant for every application. The mean load is 0.7Mb/s for e-mail, 3.3Mb/s for FTP, 1.9Mb/s for HTTP, and 25.5 Mb/s for DB Query – totaling 31.4 Mb/s which is comparable to that of CUNY, see figure 36. The duplex nature of all of the links used and the fact that the applications are being transmitted simultaneously is proved by the graphs.. These numbers reveal that the load is around 2 times that of the national campus network, which it is expected since NCN (6 nodes) has a little over half the number of nodes of the ICN (11 nodes). From the analysis of the graphs, one can say that the load increase is proportional to the number of nodes.

Table 15 shows the applications traffic load (bits/sec).

Statistic	Mean	Maximum	Minimum	Standard deviation
E-Mail	718,545	9,624,165	0	1,247,638
FTP	3,259,368	40,881,461	0	5,252,212
DB Query	25,523,230	36,660,139	0	10,724,047.98
HTTP	1,949,925	5,690,380	0	888,381

Table 15: ICN: Network Applications Traffic Load

One can use the same rationale used to explain the numbers shown in table 12 to explain the numbers in table 15. Note that ICN has 11,000 users instead of the 6,000 users in NCN.

11.4.1 ICN: End-To-End Delay - Links

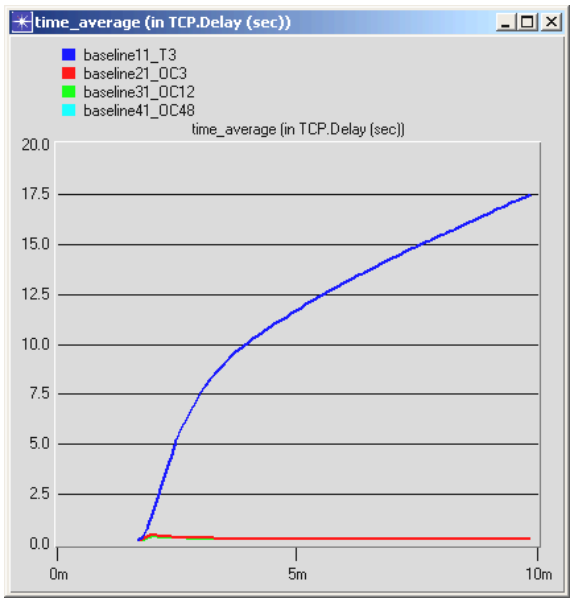


Figure 110: ICN: TCP delay

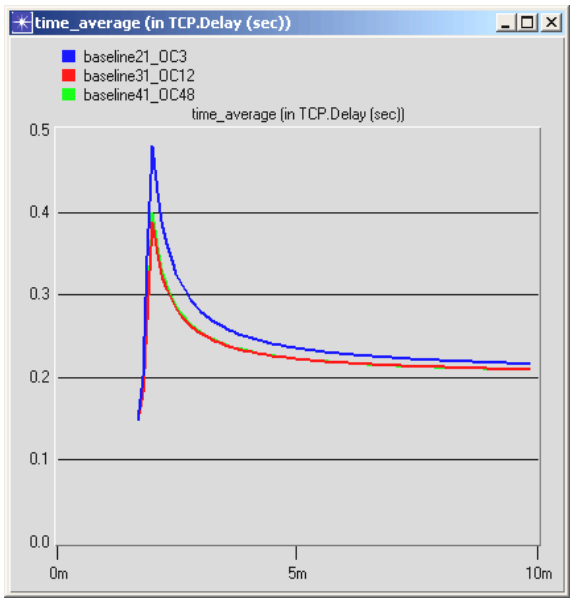


Figure 111: ICN: TCP delay – no T-3

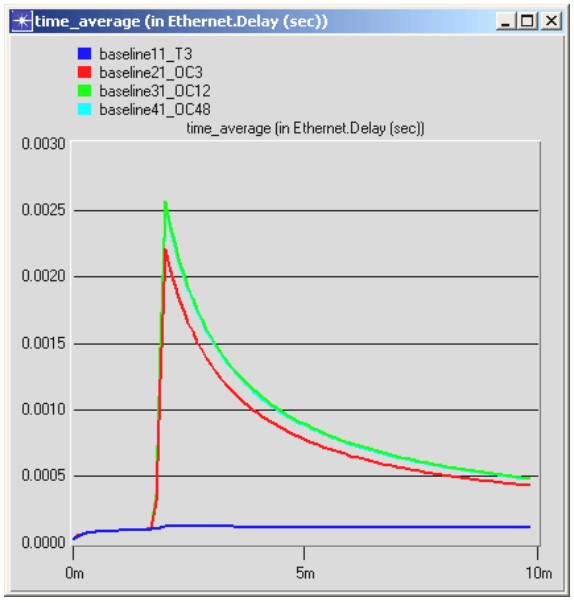


Figure 112: ICN: Ethernet delay

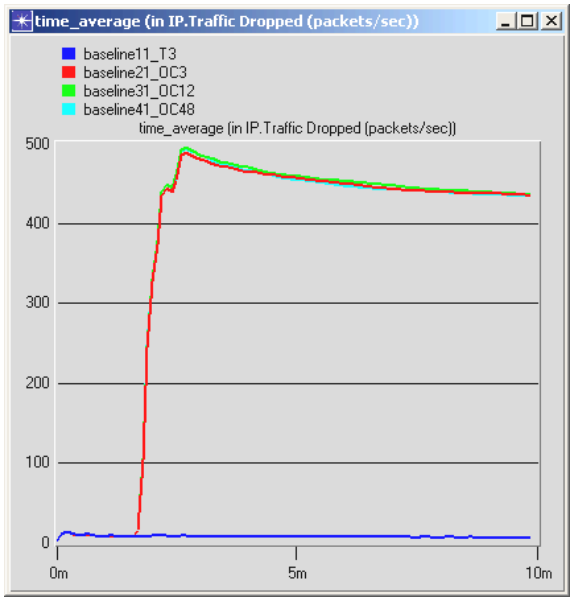


Figure 113: ICN: IP Packets Dropped

. The amount of traffic generated in the ICN oversaturates T3 link (bandwidth of 45 Mbps). Therefore the TCP delay for T3 goes up reaching 17.5 seconds at the end of the simulation run. Compare this with the value for the NCN where TCP delay for T3 was a steady 0.25 seconds. TCP delay for the optical cables shows the typical drop-tail feature and the values reaches a constant of 0.21 seconds, a little shorter than the value for NCN which was at 0.25 seconds. This demonstrates that doubling the traffic does not necessarily means doubling the delay, on the contrary, TCP delay for the optical cables is practically insensitive to this change.

. When the network reaches the steady state (to a certain degree for we can not predict what it would be the network parameter behavior past our 10 minutes of simulation run) - towards the end of the simulation run -, the Ethernet delay of 0.5 milliseconds for the optical cables is the same as it was for the NCN with had half of traffic. For T3, the Ethernet delay is a constant 0.12 milliseconds, which is about half of the delay shown in the NCN. This shows that the Ethernet switch latency is able to cope efficiently with the traffic being doubled.

. The number of IP packets being dropped shows a tremendous increase from a low average of 2 packets for NCN to an average of 435 packets for ICN. These are incoming packets that the router's queue is dropping until the queue length diminishes and the packets can be transmitted. TCP uses the fast retransmit and fast recovery to retransmit these dropped packets or some of them (in case of lost ACK).

11.4.2 ICN: Response Times (sec) - Links

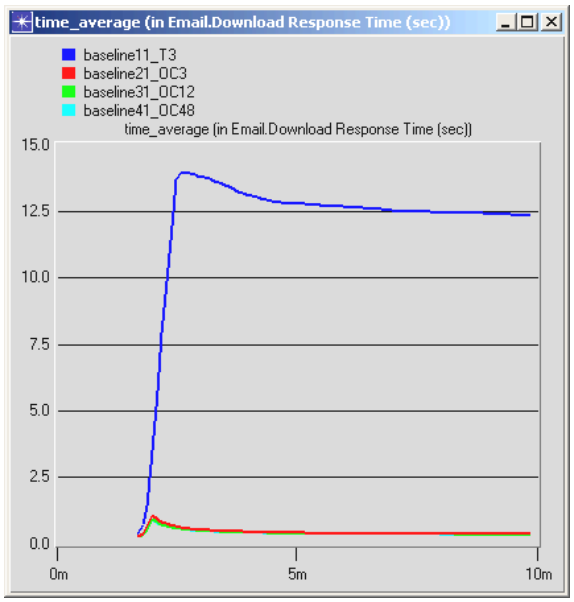


Figure 114: ICN: E-Mail

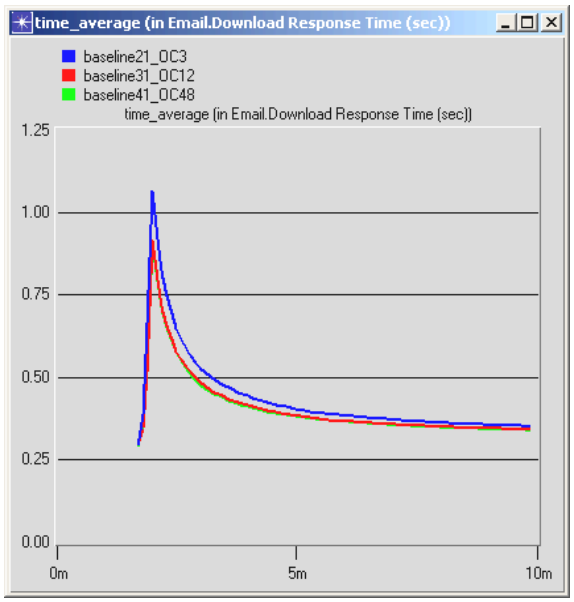


Figure 115: ICN: E-Mail – no T-3

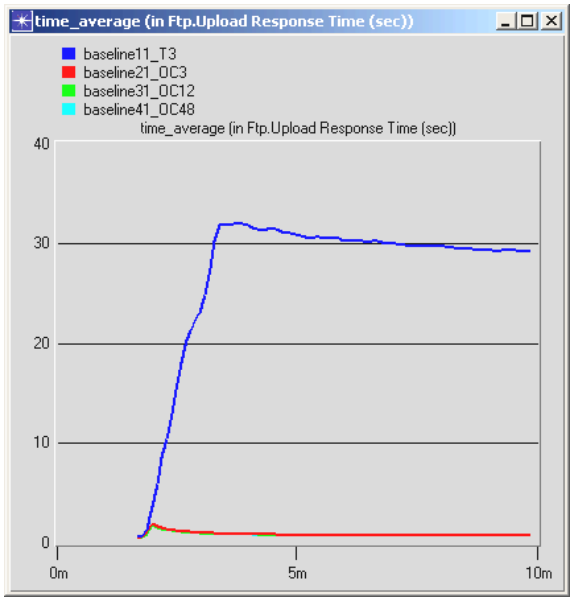


Figure 116: ICN: FTP

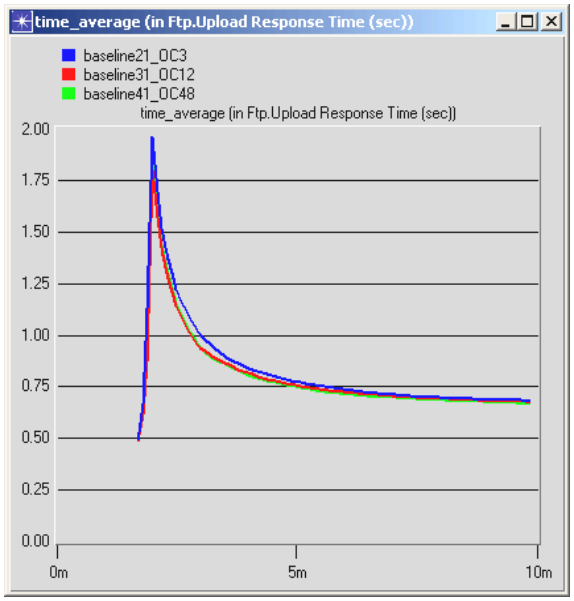


Figure 117: ICN: FTP – no T-3

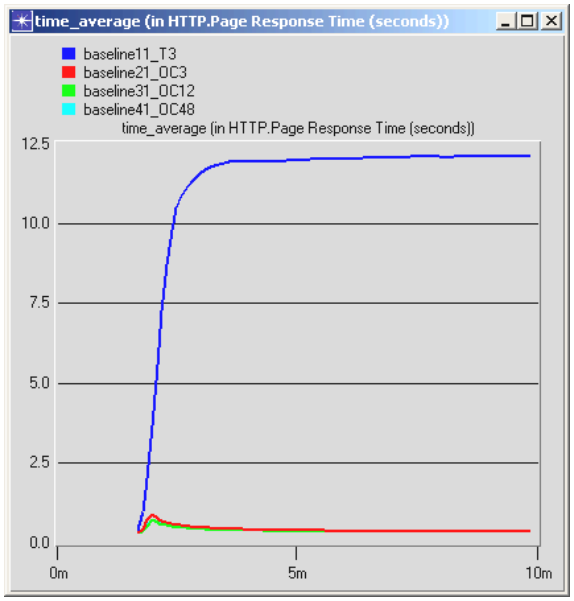


Figure 118: ICN: HTTP

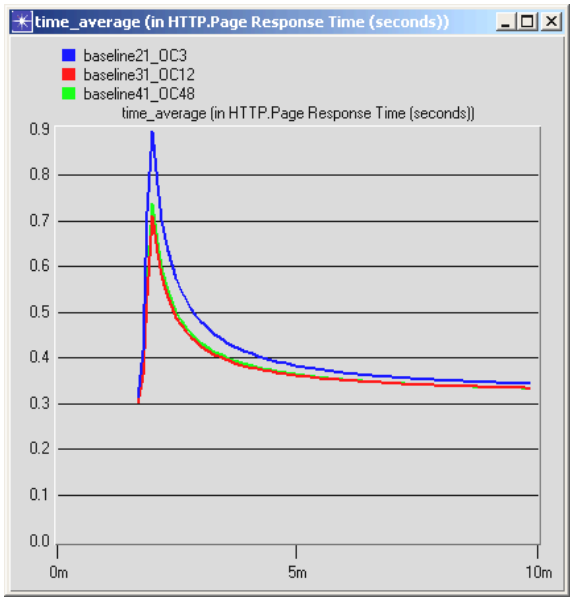


Figure 119: ICN: HTTP – no T-3

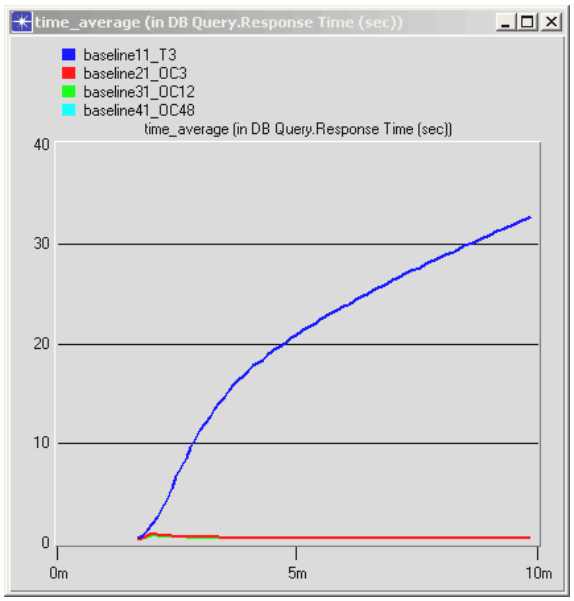


Figure 120: ICN: DB Query

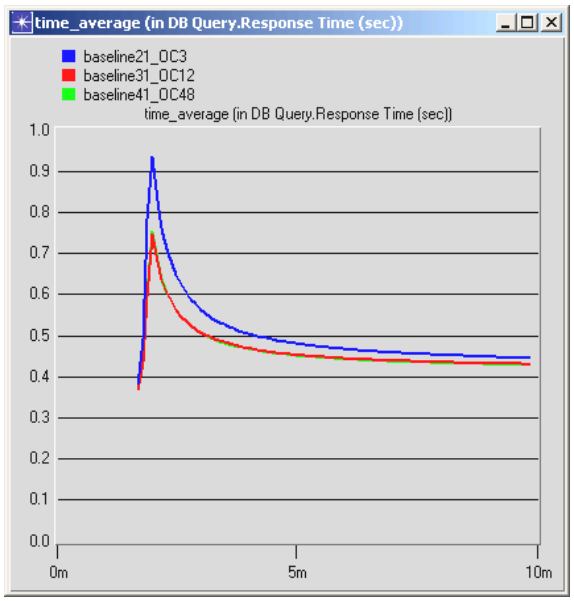


Figure 121: ICN: DB Query – no T-3

. Since the amount of traffic generated by the applications is higher than the T3 bandwidth, the response time of any application goes way up for this link.

A comparison between the mean response time for all applications for NCN and ICN is shown in table 16.

	T3			**OC3/OC12/OC48**	
	NCN	ICN		NCN	ICN
E-Mail	1.03	12.07		0.23	0.43
FTP	1.96	27.08		0.45	0.83
HTTP	0.78	11.27		0.19	0.41
DB Query	0.89	21.15		0.28	0.50

Table 16: Applications Response Times (sec)

. T-3 link is not appropriate for the high-load traffic of the ICN. Measures of the response times turn to be extremely high, notice that FTP response time reaches close to half a minute – and that's not acceptable.

. In the ICN, the optical cables result in the same response times for each application, independently of which one of the optical cables is used. Furthermore, the response times for ICN is practically double of that for the NCN, and this reflects the fact that the ICN traffic is twice that of the NCN.

11.4.3 ICN: Queue Delay - Links

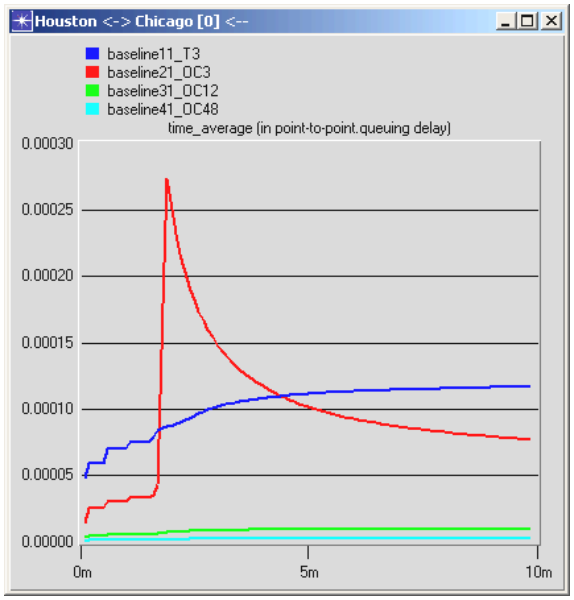


Figure 122: ICN: Queue delay Hou-Chi

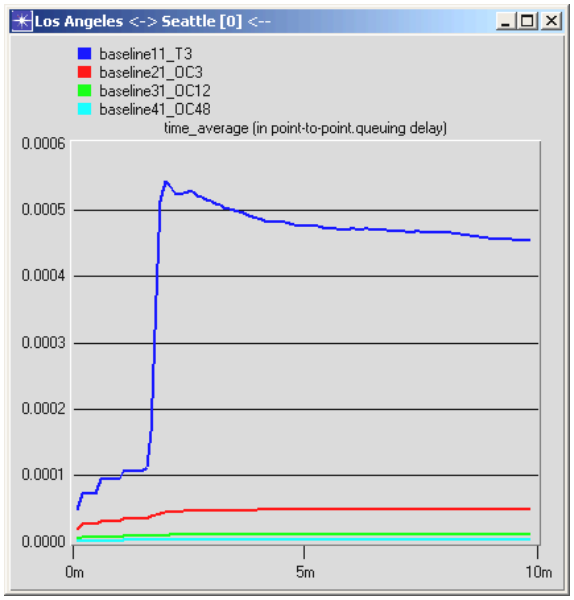


Figure 123: ICN: Queue delay La-Sea

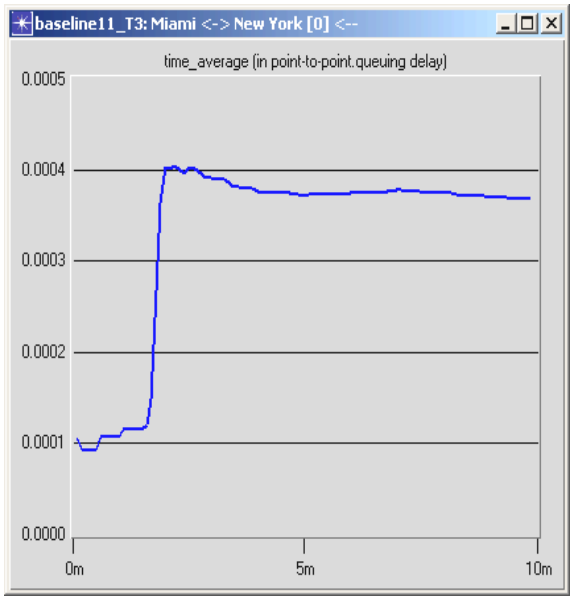


Figure 124: ICN: Queue delay NY-Mia – T-3

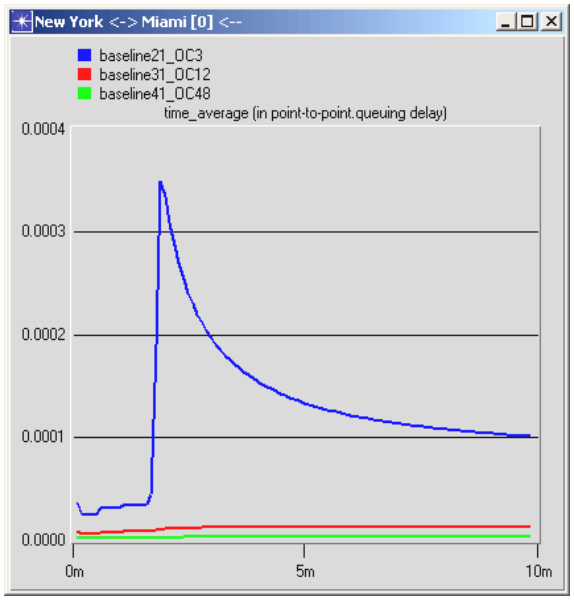


Figure 125: ICN: Queue delay NY-Mia – no T-3

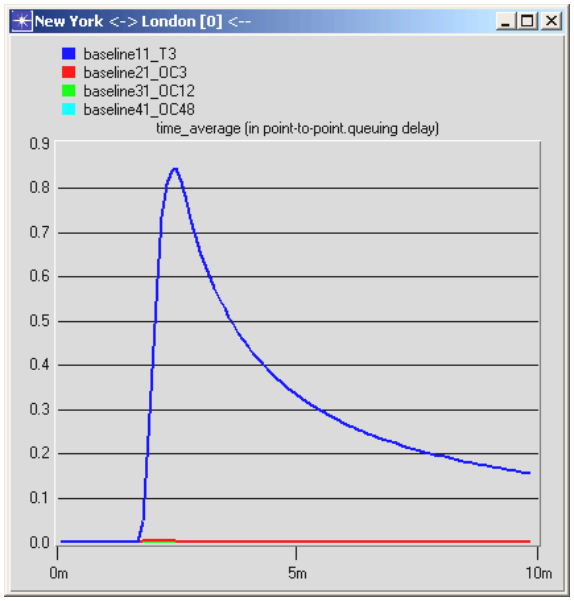


Figure 126: ICN: Queue delay NY-London

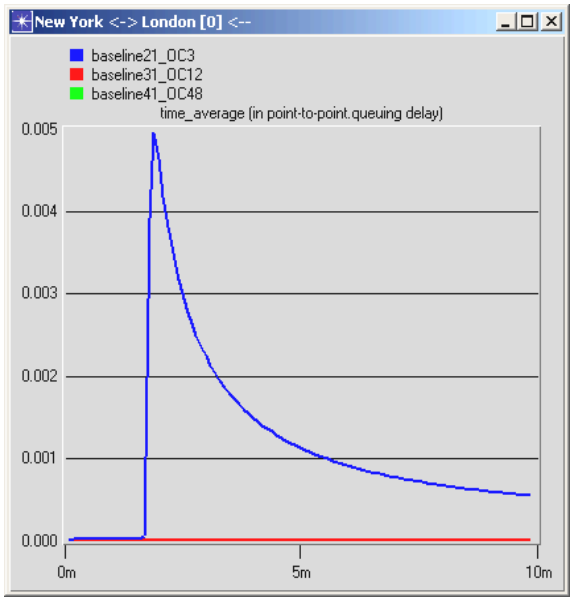


Figure 127: ICN: Queue delay La-Sea

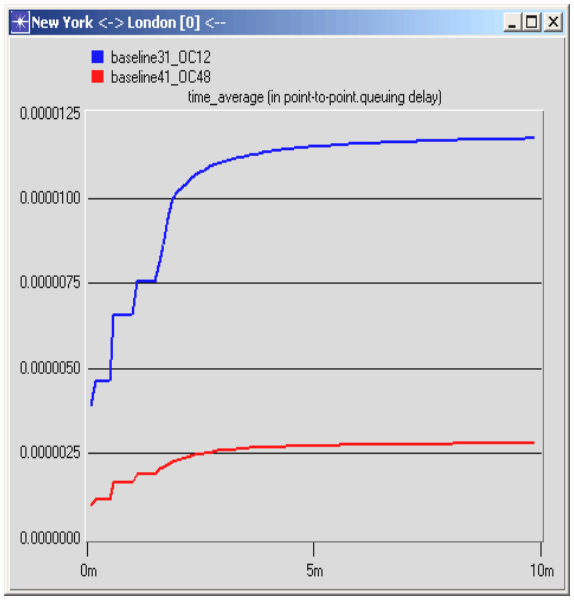


Figure 128: ICN: Queue delay NY-London noT3/OC3

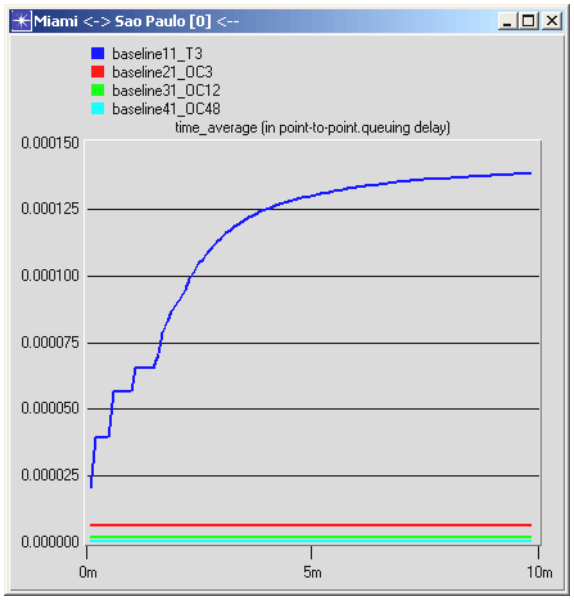


Figure 129: ICN: Queue delay Mia-SP

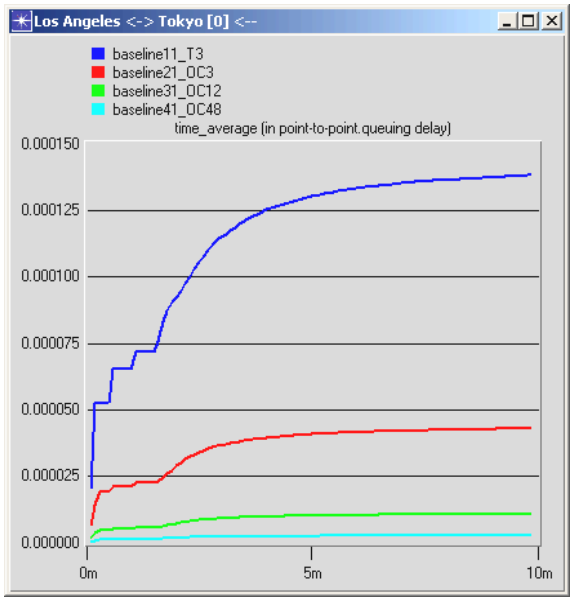


Figure 130: ICN: Queue delay LA-Tokyo

. One interesting finding of the above graphs is that the queue delay for the domestic subnet pairs are the same for NCN and ICN for each type of cable used to connect the pair. That is, even though the traffic for ICN is twice that of NCN, the queue delay for each link used is the same for both configurations. Distinction has to be made for the T3 link with a queue delay of under 0.5 ms, while the optical cables present a queue delay ranging from 0.07 milliseconds for Houston-Chicago to 3 microseconds for NY-Miami.

. For the other pairs of subnets (NY-London, Miami-Sao Paulo, and LA-Tokyo), the queue delay reaches 300 milliseconds for the NY-London using T3, and then drops to units of microseconds for the optical cables; it is less than a millisecond with T3 connecting the two other subnet pairs, and then it is becomes microseconds for the OC-cables.

. 11.4.4 ICN: Throughput - Links

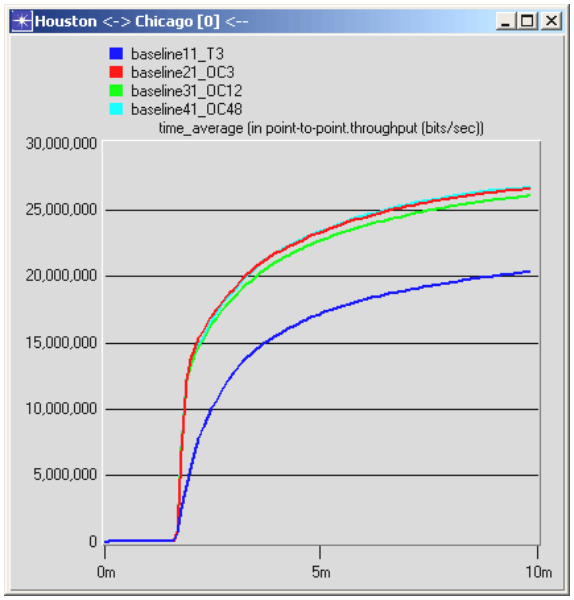


Figure 131: ICN: Throughput Hou-Chi

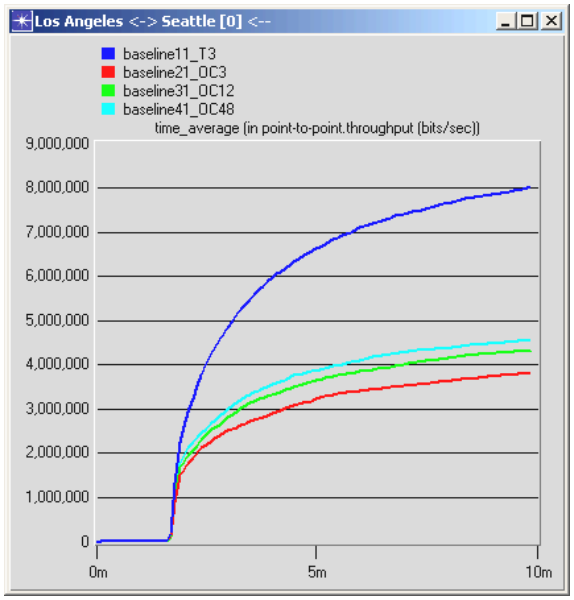


Figure 132: ICN: Throughput La-Sea

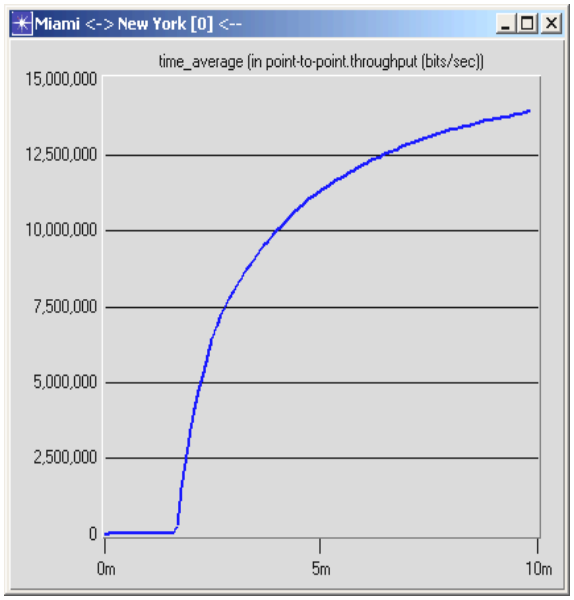


Figure 133: ICN: Throughput NY-Mia – T-3

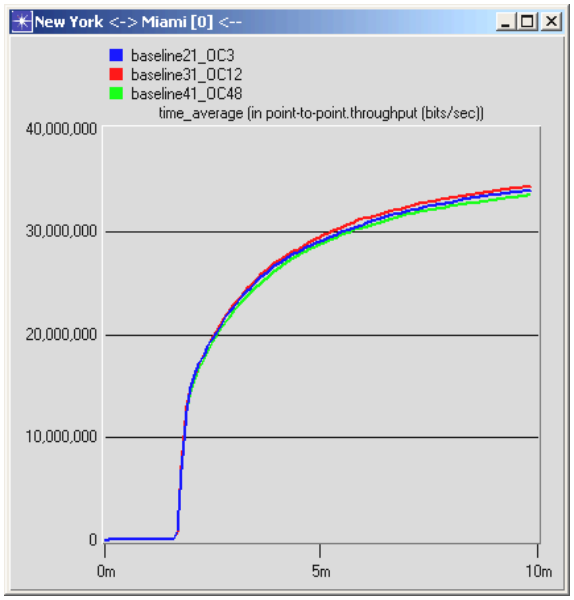


Figure 134: ICN: Throughput NY-Mia – no T-3

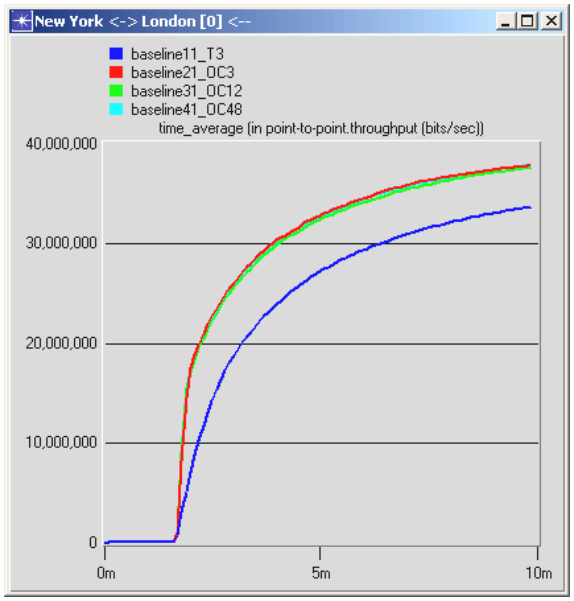


Figure 135: ICN: Throughput NY-London

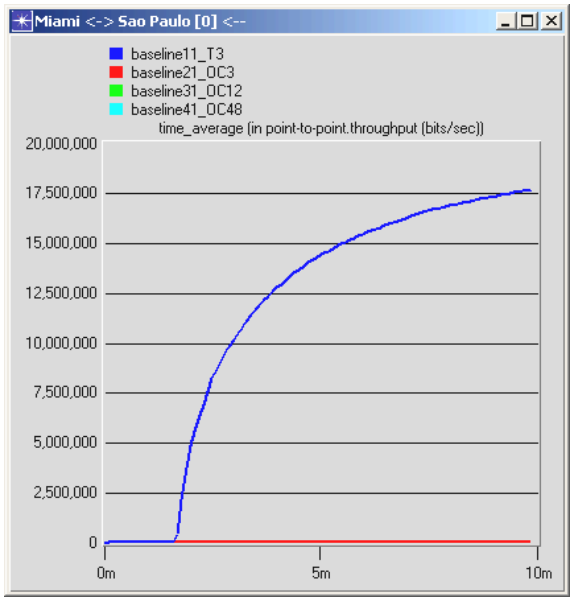


Figure 136: ICN: Throughput Miami-SP

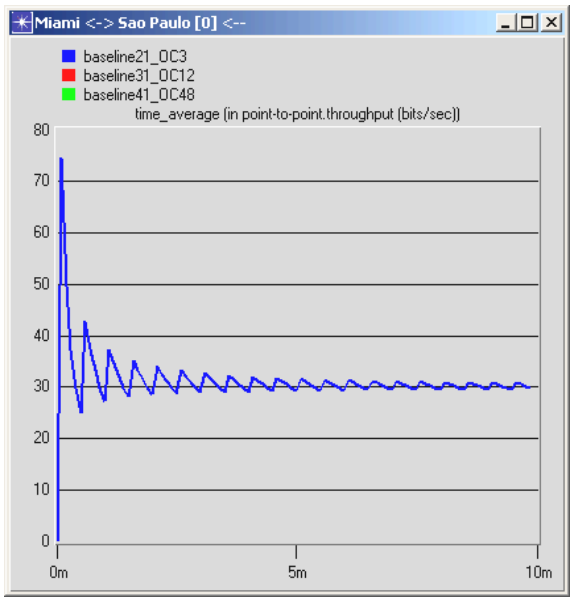


Figure 137: ICN: Throughput Mia-SP – no T-3

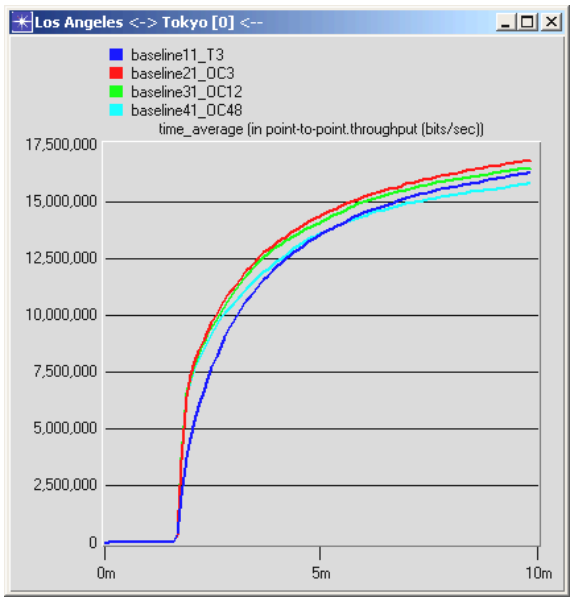


Figure 138: ICN: Throughput LA-Tokyo

. Throughput present with following features:

. As it was with NCN, the throughput in each subnet pair being considered is practically the same for any of the links used, independently if it is digital or optical. Although there is a minor discrepancy when T3 is being used to connect LA-Seattle and Miami-Sao Paulo. Measured throughput in a specific link can not exceed the link's bandwidth, thus for example for Miami-Sao Paulo while the throughput is about 60 bps when using any of the optical cables, it measures 36 Mbps for T3 which has a bandwidth of 45 Mbps.

Table 17 shows a comparison between the mean throughputs in NCN and ICN for the selected subnet pairs and links.

	NCN	**ICN**
Houston-Chicago	31	53
LA-Seattle	8	9
NY-Miami	23	65
NY-London		74
Miami-Sao Paulo		36
LA-Tokyo		33

Table 17: ICN Subnet Throughput (Mbps)

11.4.5 ICN: Utilization - Links

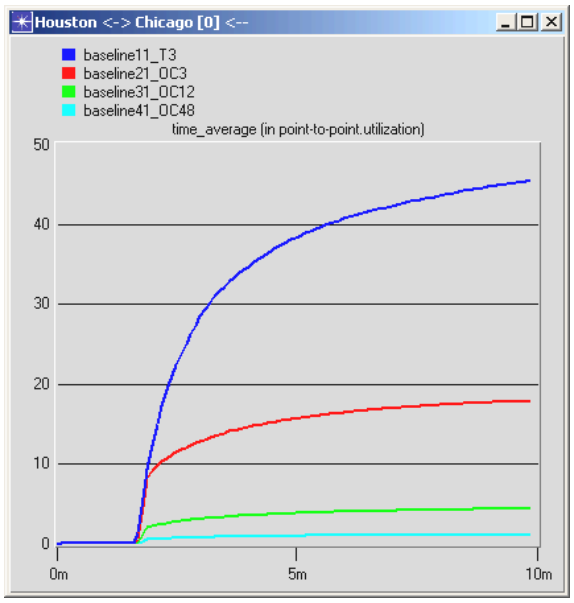


Figure 139: ICN: Utilization Hou-Chi

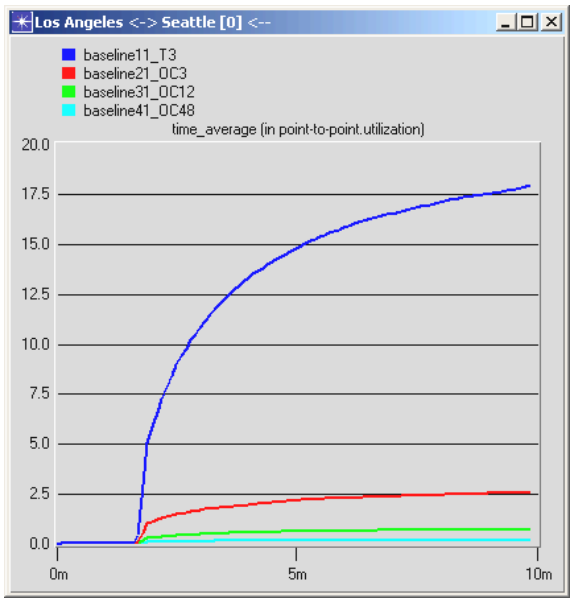


Figure 140: ICN: Utilization LA-Seattle

Figure 141: ICN: Utilization NY-Miami – T3

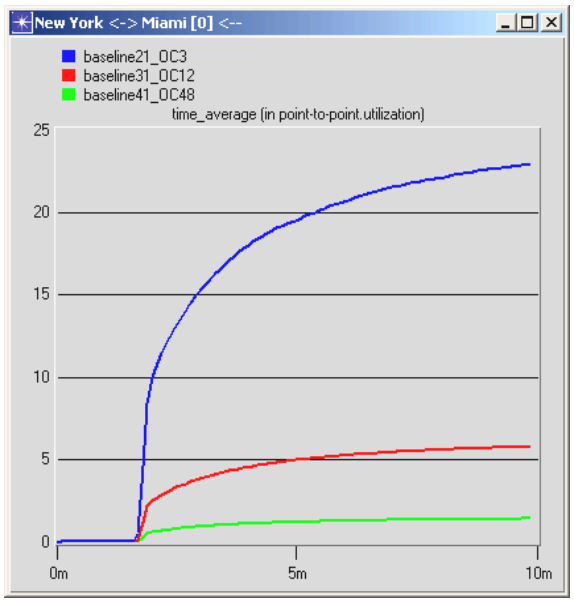


Figure 142: ICN: Utilization NY-Miami – no T3

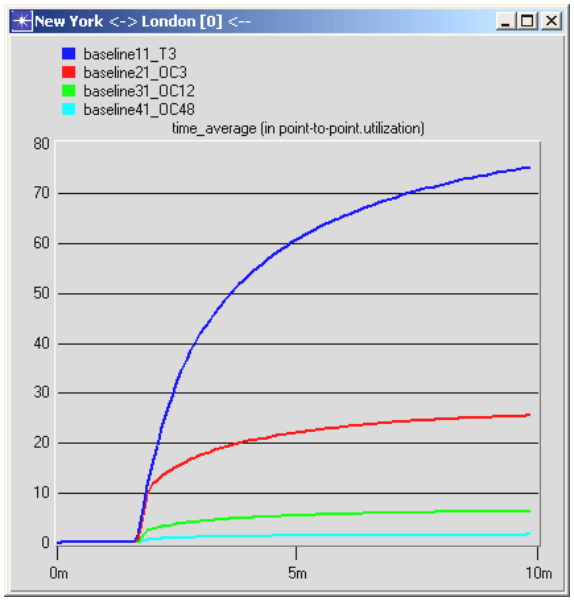


Figure 143: ICN: Utilization NY-London

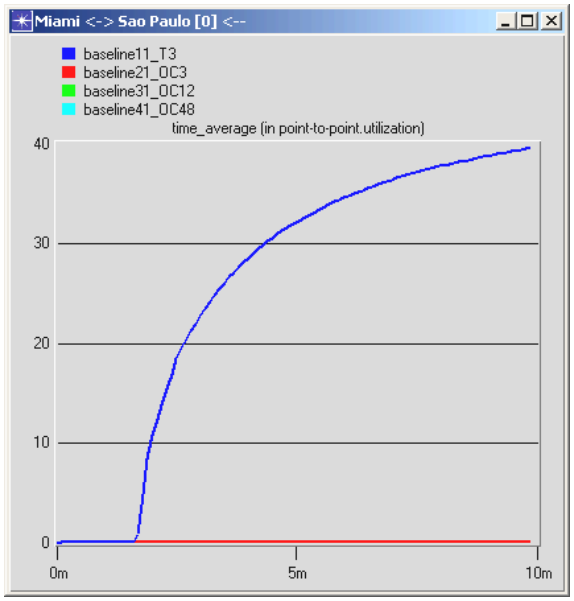


Figure 144: ICN: Utilization Miami-SP

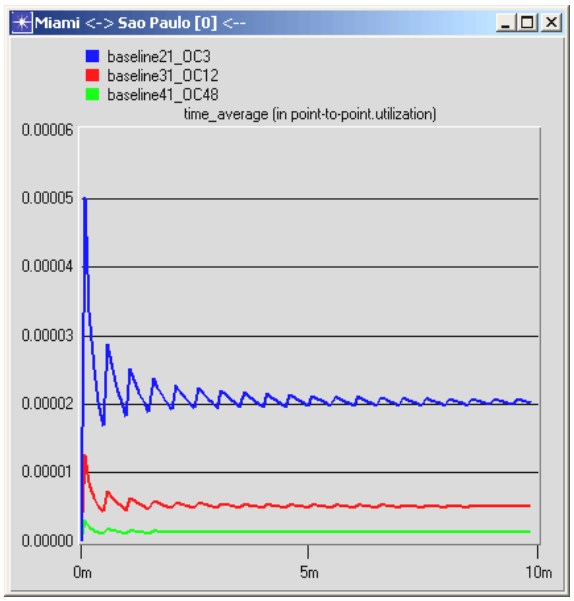


Figure 145: ICN: Utilization Miami-SP no T3

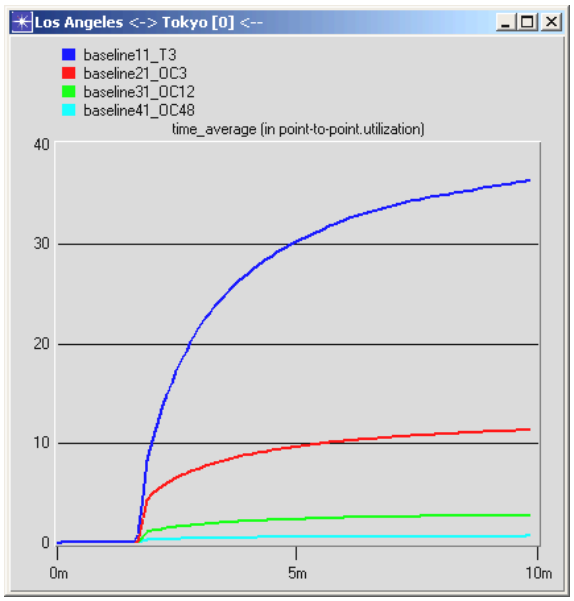


Figure 146: ICN: Utilization LA-Tokyo

. The table 18 shows the link utilization for the selected subnet pairs and links. Note that the utilization of T3 link is oversaturated for NY-London, and it is close to capacity for Houston-Chicago, Miami-Sao Paulo, and LA-Tokyo.

	T-3	**OC-3**	**OC-12**	**OC-48**
Houston-Chicago	91	36	9	2
LA-Seattle	36	5	1	0.5
NY-Miami	62	46	12	3
NY-London	151	51	13	3
Miami-Sao Paulo	79	4	10	3
LA-Tokyo	73	23	6	1

Table 18: Link Utilization

. Note that for the NCN, 72% of the T3 bandwidth was being utilized in the Houston-Chicago link, and close to 50% utilization of its bandwidth in the NY-Miami link. These facts demonstrate that T-3 is not an appropriate link for our interconnected campus networks even for a short-term network growth – campus network designers expect demand for network bandwidth to grow from 10 to 20% yearly [32]. For our networks, T-3 could be used for selected links, such as the one connecting LA-Seattle.

11.5 ICN: Scenario 23 – OC-3 Link + Backload + QoS

Following the same method used previously in the National Campus Network (NCN), two new factors are introduced into the international campus network (ICN): background traffic and quality of service. Links carry traffic generated by our applications and traffic generated by extraneous sources. Also, traffic are now classified according to the type of application they were generated from. Routers are programmed to differentiate this traffic and proceed in accordance to the rules of the Diffserv mechanism of quality of service. Buffer size for database query traffic are increased from 500 to 1,000 bytes to further expedite the processing of these packets with minimum or no packets being discarded according to the RED schema. The other applications remain with a buffer size of 500 bytes.

Under a hypothetic situation where users have at their disposal knowledge data bases across the interconnected campuses, the idea is to establish a mechanism through which the access to these knowledge bases is set to a higher priority during the process of forwarding network traffic. This could the basis of an actual distributed knowledge network where quality of service is employed to maximize the exchange of education in any area, i.e., domain. Traffic differentiation can then be assigned according to the specific education domain that is stored and updated in each knowledge base. For this to be implemented further inspection of the packet needs to be available in order for a network interface to discern what education domain the packet belong to. In reality, the fundamentals of intelligent interface for this purpose are already being carried out, and its

prototype is being hailed as a landmark intelligent network peripheral [33]. The only obstacle in using this prototype is that it is not yet included on any commercial simulator.

The previous graphs showed that T-3 link results in very low network performance as shown by the high delays for TCP transmissions and very high response times for every one of the applications.

In this scenario we have the distributions for the OC-3 link with separated runs of link alone, link plus background traffic, and link plus background traffic and QoS. The reason for analyzing exclusively OC-3 distributions is because we could finish within a reasonable time (over 23 hours!) the simulation run when the background traffic is included with the explicit traffic. The computing resources at hand were not able to complete the background traffic runs for the other optical cables. However, the simulations were possible when QoS was present, which will be analyzed in the next scenario.

Regarding T-3, none of the simulations were completed, which it was predictable since the previous section showing our ICN analysis for the undisturbed link revealed oversaturated link for most of the network. We had a close to five minutes run for the background traffic, and then the simulation was stopped with the simulation status graph showing no event being processed. Because of the load in the network there would not be any change if we had used a more powerful computer – as it was concluded before, T-3 is not appropriate to be the link connecting all of the subnet pairs in the campus networks.

11.5.1 ICN: End-To-End Delay – (OC-3) + Backload + QoS

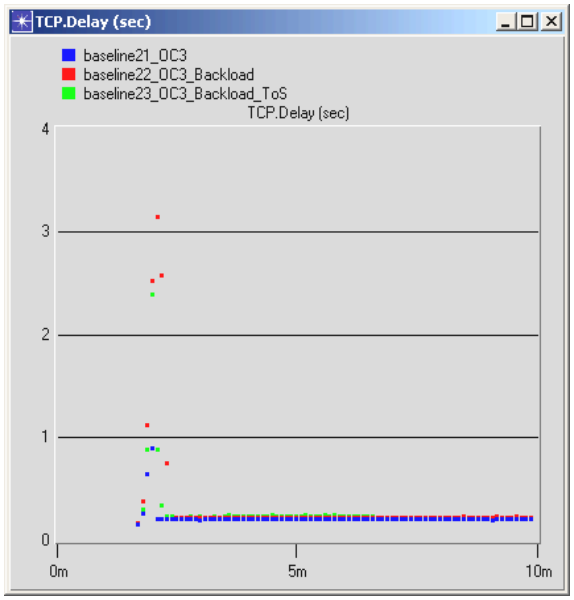


Figure 147: ICN: TCP delay (as is)

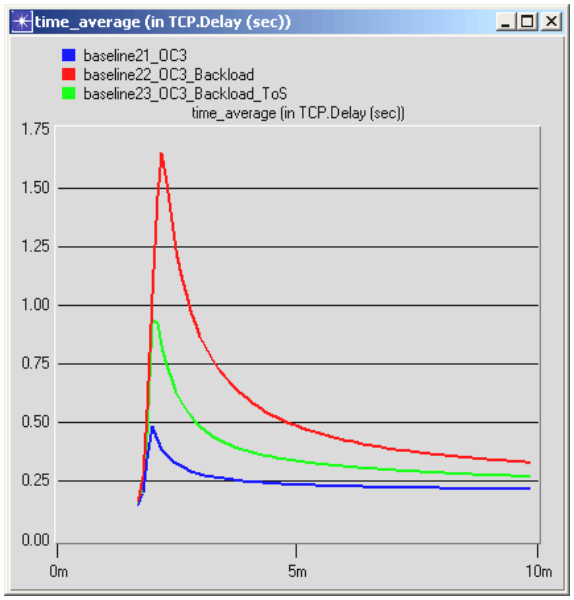


Figure 148: ICN: TCP delay

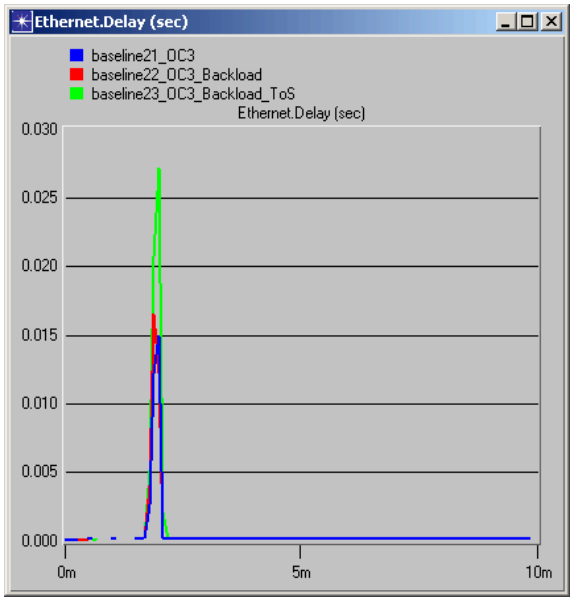


Figure 149: ICN: Ethernet delay (as is)

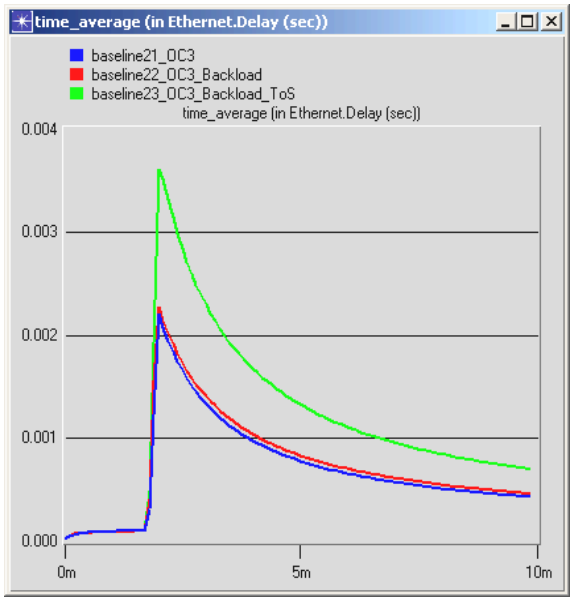


Figure 150: ICN: Ethernet delay

. We deliberately showed the global delays graphs in the “as is” and in the “time average” formats for comparison purposes. The “as is” for the TCP delay consists initially of isolated dots reflecting the fact that each delay point is associated with one or more applications. While the initial graph of Ethernet delay consists of solid line due to a continuous measure of the end-to-end delay of every packet.

. The mean values of TCP delay for link, link plus backload, and link plus backload and QoS are in milliseconds: 220, 330, and 260. Therefore, the same behavior shown in the NCN is also present now, that is, the background traffic increases the delay, and the delay is then reduced when QoS is introduced in the network. This is evidence of the valuable contribution of QoS to the network performance, because the ultimate goal of network managers (and users) is to have a network with high bandwidth, and low delay and response time.

. The Ethernet delay for its typical diminutive values is often referred to in microseconds terms. The graph for Ethernet delays shows the following values for link, link + backload, and link + backload + QoS: 400, 500, and 700. The difference now is that QoS does not positively influence the delay, on the contrary, it augments it. This is due to the fact that the end-to-end delay is calculated for each packet and QoS is actually adding an extra amount of delay when packets are going through the routers, after all Ethernet delay is not interested in this process.

11.5.2 ICN: Response Times – (OC-3) + Backload + QoS

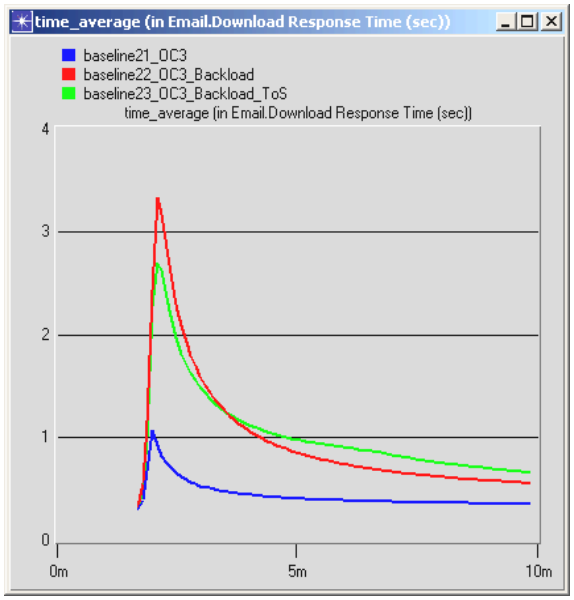


Figure 151: ICN: E-Mail

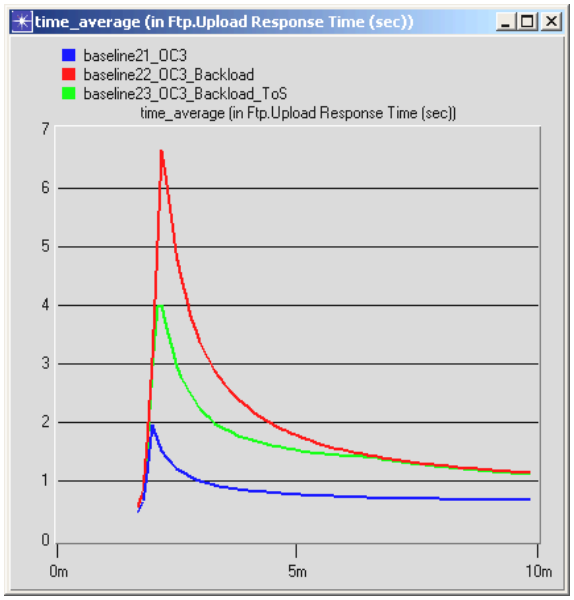


Figure 152: ICN: FTP

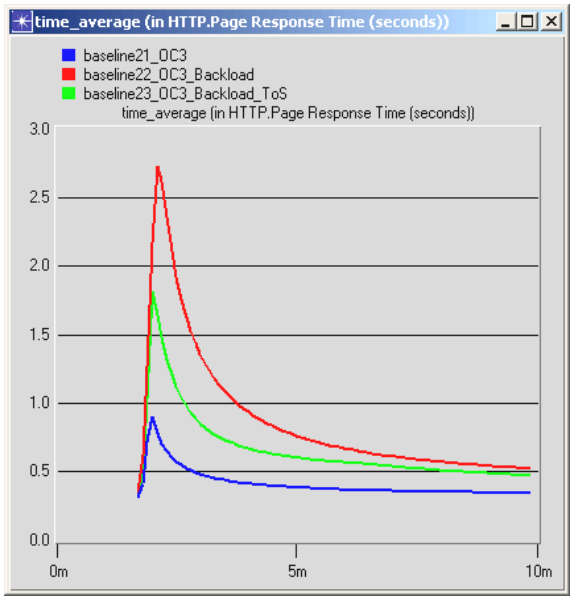


Figure 153: ICN: HTTP

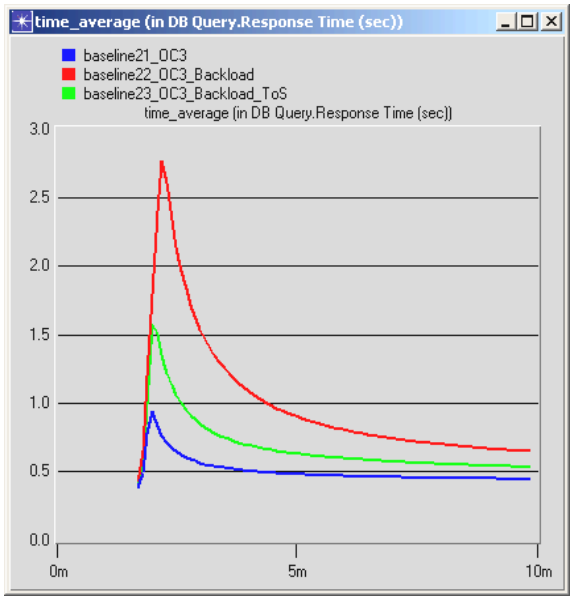


Figure 154: ICN: DB Query

. While for the NCN the response times were not affected by either background traffic or QoS due to the amount load when using OC-3 link, for ICN the graphs above clearly show an improvement in the response times when the hybrid traffic (explicit + background traffic) includes QoS compared with the hybrid traffic without QoS . The numbers reveal a 20% reduction of response time for DB Queries (recall that this application is assigned the highest priority), 10% shorter response for HTTP, and 3% lower response time for file transfers. The e-mail graph shows n unexpected 13% increase in the response time, however a closer look at the graph reveals a distinct tendency of the backload distributions with or without QoS to converge, feature such that is present in all the others "Best Effort" applications, with exception of DB Queries distributions. This convergence is a result of the fact that the corresponding packets of the "Best Effort" applications have the same priority.

. Table 19 lists all the applications response times (in milliseconds) for ICN and NCN.

	ICN				**NCN**
	OC-3	**OC-3 + Bkld**	**OC-3 + Bkld + QoS**		**L+B+Q**
E-Mail	350	550	620		200
FTP	680	1140	1110		360
HTTP	340	520	470		170
DB Query	450	650	520		260

Table 19: Applications Response Times (ms) for NCN and ICN

11.5.3 ICN: Queue Delay – (OC-3) + Backload + QoS

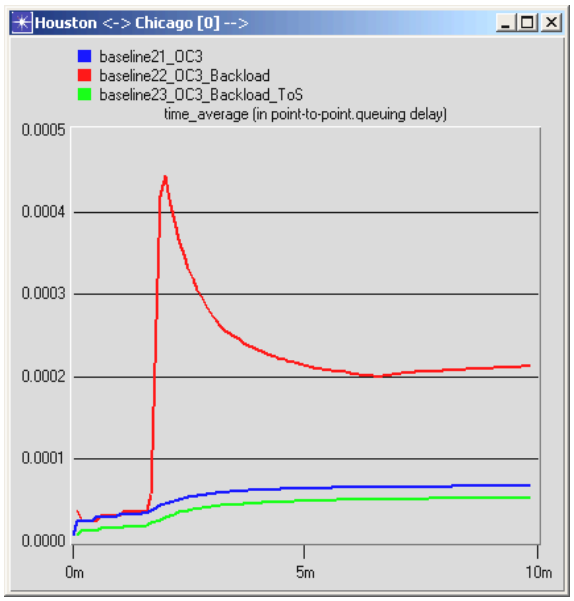


Figure 155: ICN: Queue delay Hou-Chi

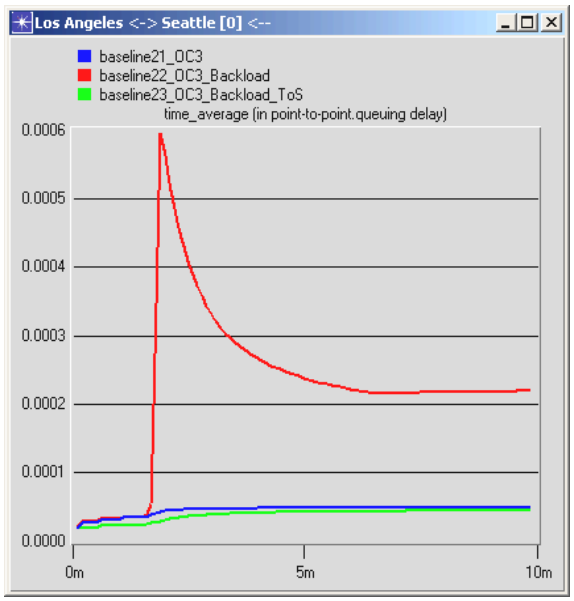


Figure 156: ICN: Queue delay La-Seattle

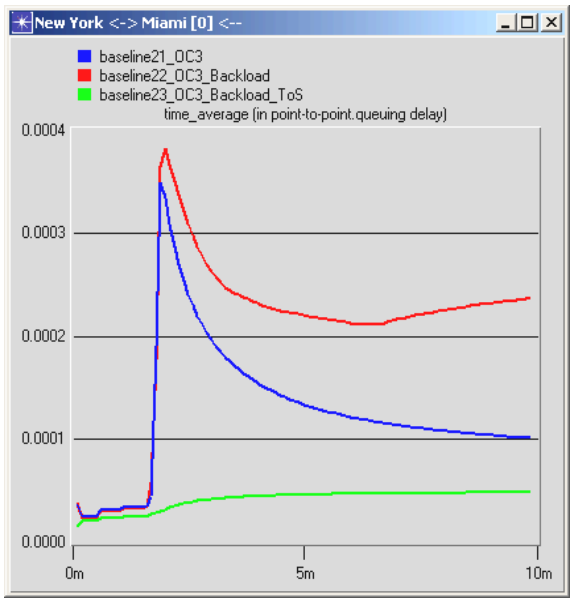


Figure 157: ICN: Queue delay NY-Miami

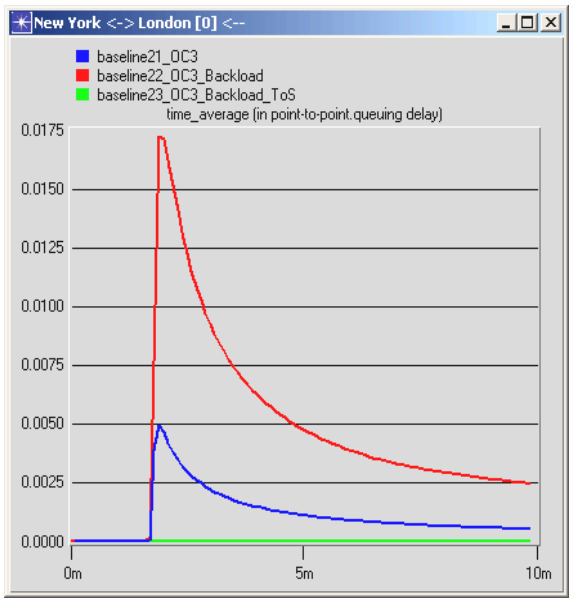


Figure 158: ICN: Queue delay NY-London

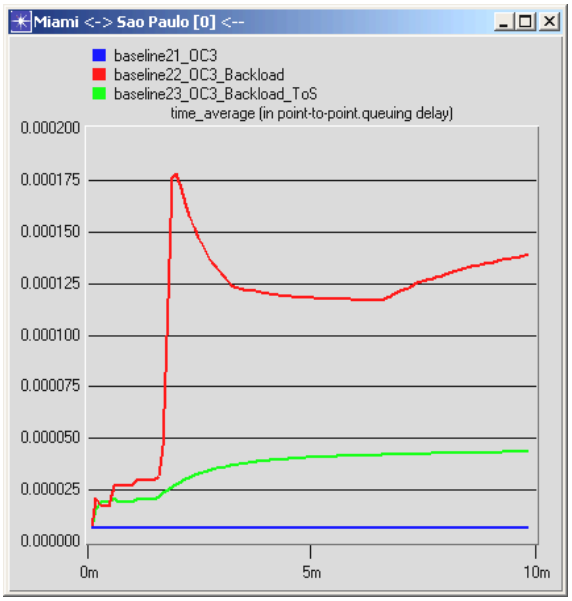


Figure 159: ICN: Queue delay Miami-SP

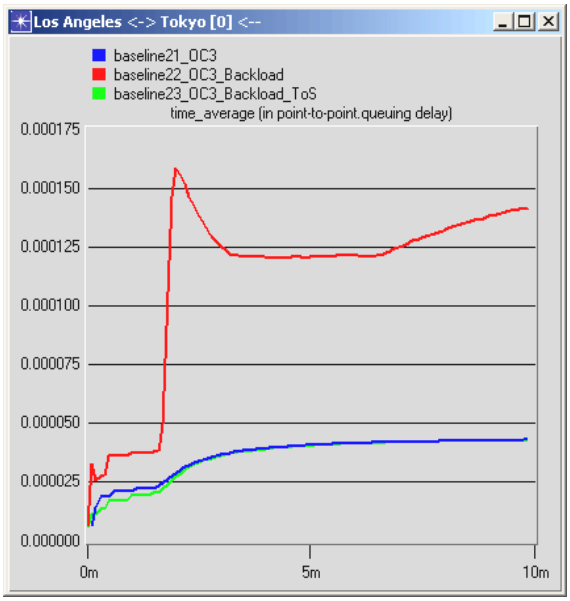


Figure 160: ICN: Queue delay LA-Tokyo

. The queue delay increases when the background traffic is injected into the network, then the delay is reduced by the QoS, and this is evident from the graphs for all the links connecting the selected subnet pairs. This corresponds to the performance of FIFO mechanism in the routers and also the WFQ mechanism when QoS is present in the network. Again, this demonstrates the efficiency of QoS by expediting the traffic through the QoS-enabled routers.

. QoS reduces the queue delay for the hybrid traffic by an average of 80% for all the subnets links - except Houston-Chicago where the delay is reduced by 40%. Compare this value with those for the NCN, where QoS reduced the queue delay by 55% including Houston-Chicago. Note that Houston-Chicago link represents approximately the mid-way link of the entire network, be it NCN or ICN, and therefore all of the traffic that runs into

one link of the network that connects to either Houston or Chicago is branched to two flows: one that continues on over the same link and the other one that goes over the Houston-Chicago link.

. Table 20 shows the queue delays (in milliseconds) for the pairs of subnets for ICN and NCN.

	ICN				**NCN**		
	OC3	**+ Bkld**	**+ QoS**		**OC3**	**+ Bkld**	**+ QoS**
Houston-Chicago	0.14	0.40	0.25		0.12	0.18	0.08
LA-Seattle	0.10	0.44	0.08		0.08	0.18	0.08
NY-Miami	0.20	0.48	0.10		0.12	0.22	0.10
NY-London	1.10	5.00	0.08				
Miami-Sao Paulo	0.02	0.28	0.08				
LA-Tokyo	0.08	0.28	0.08				

Table 20: Queue delay (ms) for ICN and NCN – OC-3 link

. One distinct feature of the graphs is that, for either NCN or ICN, the queue delay with QoS present is a steady 0.08 ms for all of the subnet pairs, and with not much dependency on the amount of traffic (ICN traffic is twice that of NCN). Once more, the use of QoS, which decreases the queue delay, is compelling in the sense that traffic management improves considerably the performance of the network.

11.5.4 ICN: Throughput – (OC-3) + Backload + QoS

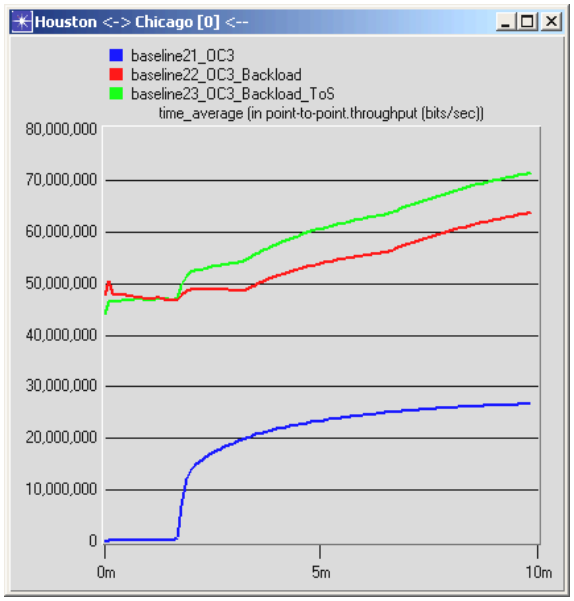


Figure 161: ICN: Throughput Hou-Chi

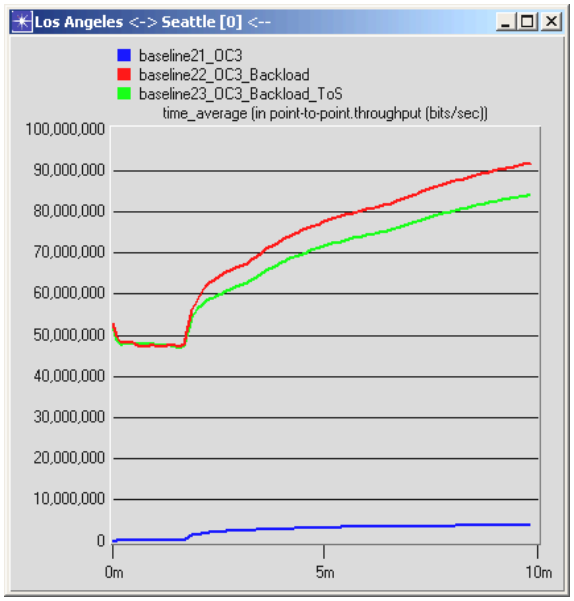


Figure 162: ICN: Throughput La-Seattle

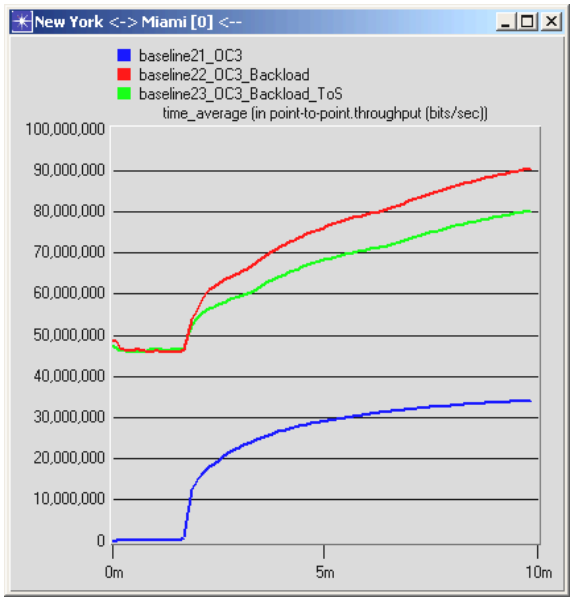


Figure 163: ICN: Throughput NY-Miami

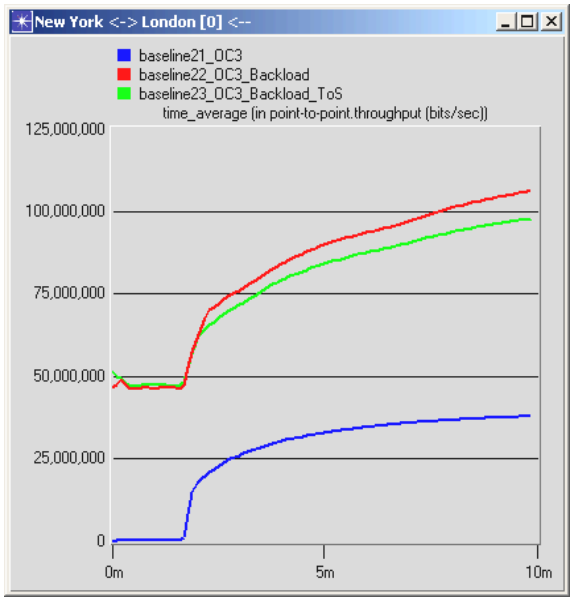


Figure 164: ICN: Throughput NY-London

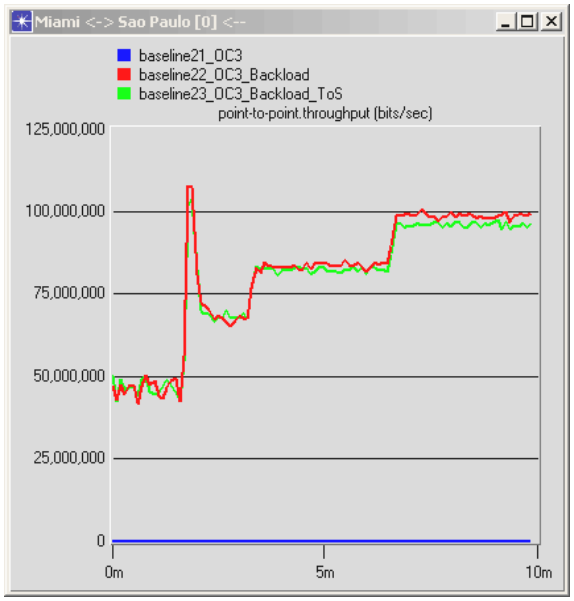


Figure 165: ICN: Throughput Miami-SP (as is)

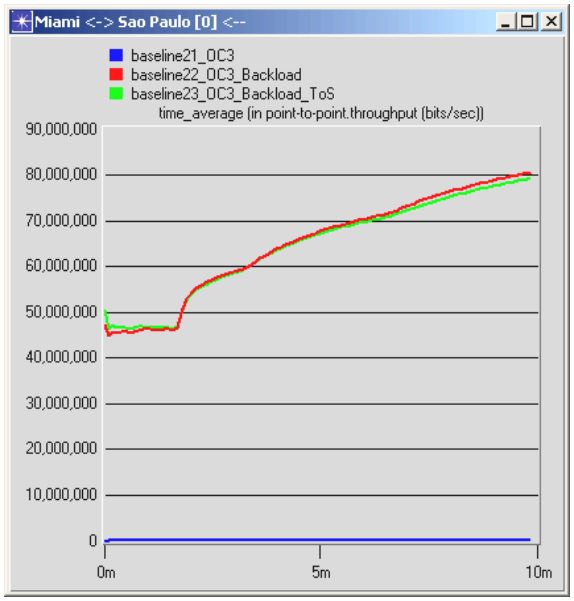


Figure 166: ICN: Throughput Miami-SP

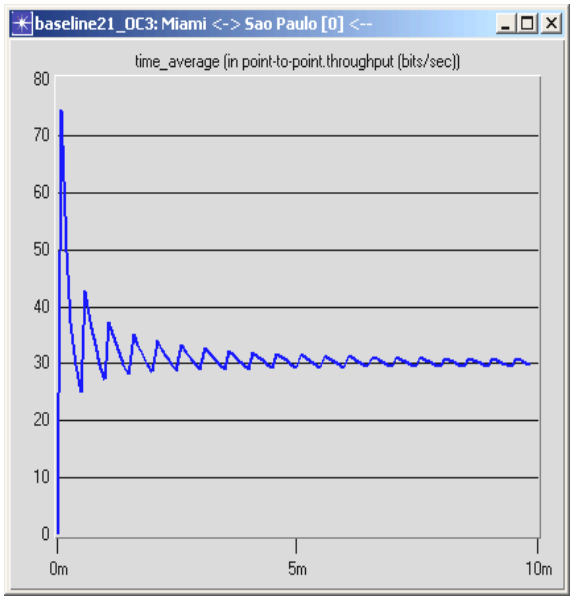


Figure 167: ICN: Throughput Miami-SP (OC3 only)

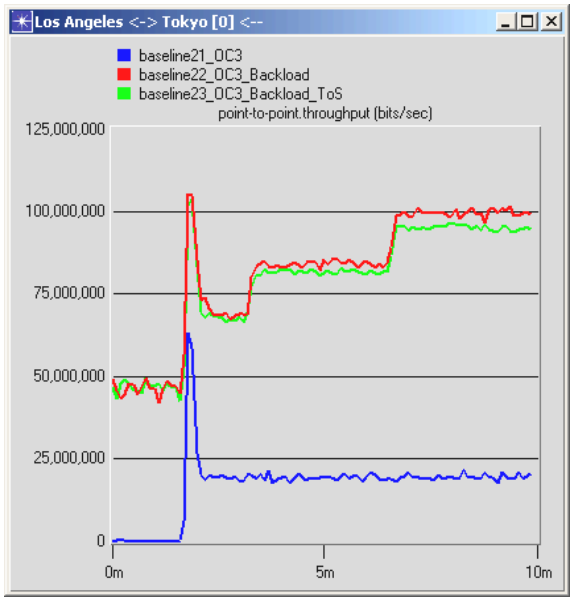


Figure 168: ICN: Throughput LA-Tokyo (as is)

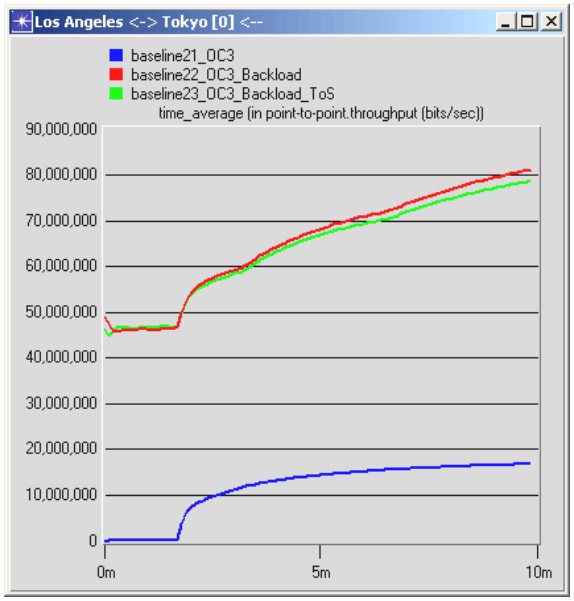


Figure 169: ICN: Throughput LA-Tokyo

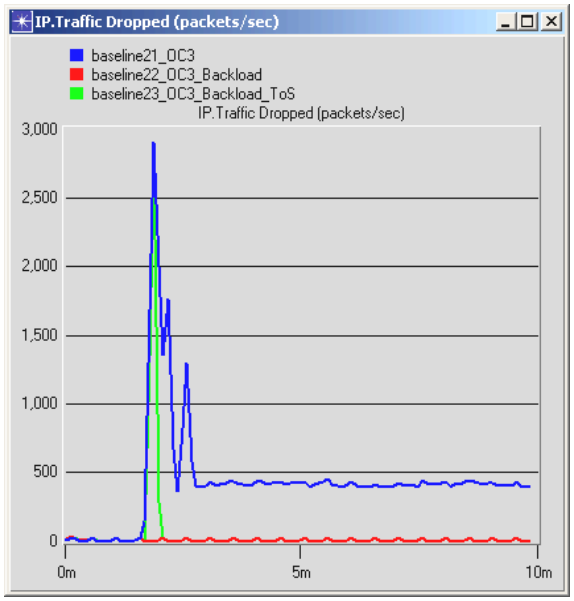


Figure 170: ICN: IP Traffic Dropped

. It is inconclusive the increase or decrease of throughput with QoS. All of the graphs shown above have the throughput in only one direction – to have the full value one needs to add the throughput for the other direction since all links are duplex links. The comfort is that almost all of the distributions in both directions are similar and therefore one needs only to double the values to get the full amount of throughput in the link. The few ones (La-Tokyo and Miami-Sao Paulo with less connections to other subnets) showing different graphs for each direction also reveal that one graph is a mirror image of the other, that is, in one graph the throughput line with QoS lies at a short distance below the throughput line with background without QoS, and in the other graph the QoS line lies at the same short distance as the first graph but now above the background line.

. The numbers reveal a 373% increase in the throughput from the undisturbed link to the link with background traffic and QoS. It is reasonable to expect a negative contribution of the QoS for the change in throughput, since a better network traffic management means an efficient flow of traffic, although with less throughput. Table 21 compares the throughput (Mbps) for ICN and NCN when the link is OC-3. Also shown are the statistics for the scenario with background traffic and for the scenario with background and QoS.

	ICN				**NCN**		
	OC-3	**+ Bkld**	**+ QoS**		**OC-3**	**+ Bkld**	**+QoS**
Houston-Chicago	29	155	143		27	144	142
LA-Seattle	22	158	172		15	143	142
NY-Miami	43	143	157		16	144	142
NY-London	81	217	199				
Miami-Sao Paulo	0.06	163	159				
LA-Tokyo	35	164	159				

Table 21: Throughput for ICN and NCN (Mbps) – OC-3 link

. Interesting findings: in the NCN (also for ICN to the certain extent) the background traffic and the QoS results in the same amount of throughputs. Also, for NCN and ICN, QoS in average decreases by 1% the throughput for the link with background.

11.5.5 ICN: Utilization – (OC-3) + Backload + QoS

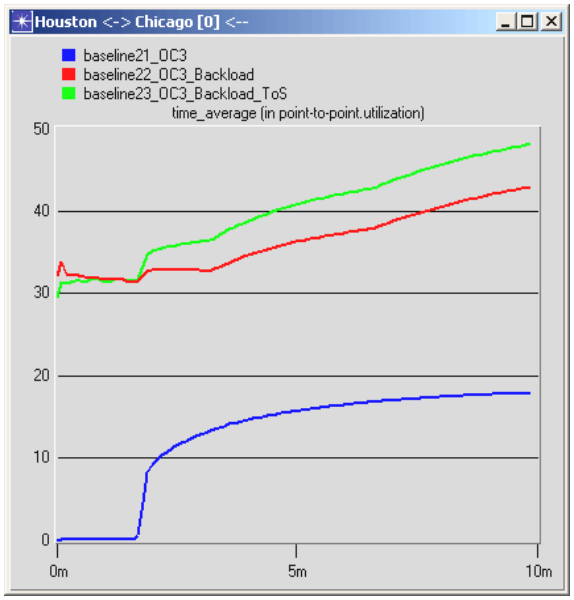


Figure 171: ICN: Utilization Hou-Chi

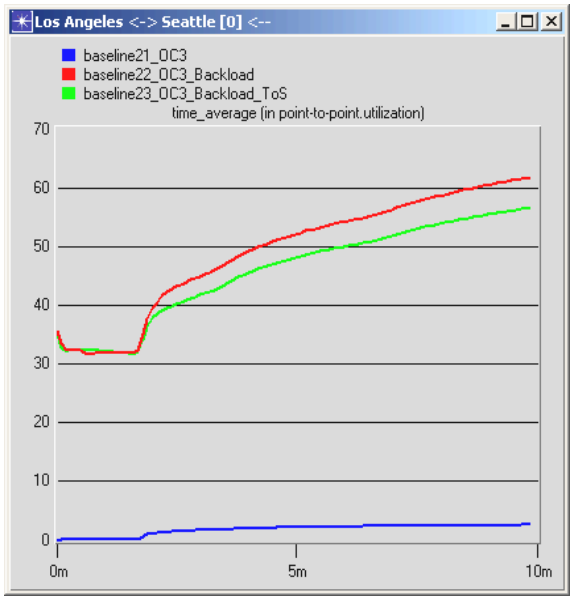


Figure 172: ICN: Utilization La-Seattle

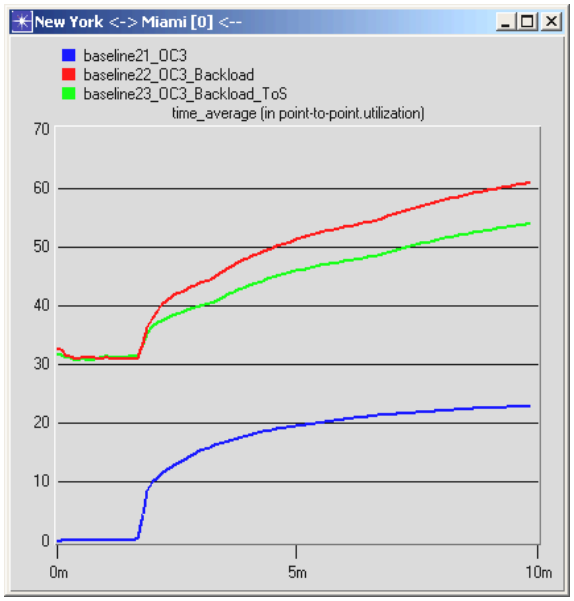


Figure 173: ICN: Utilization NY-Miami

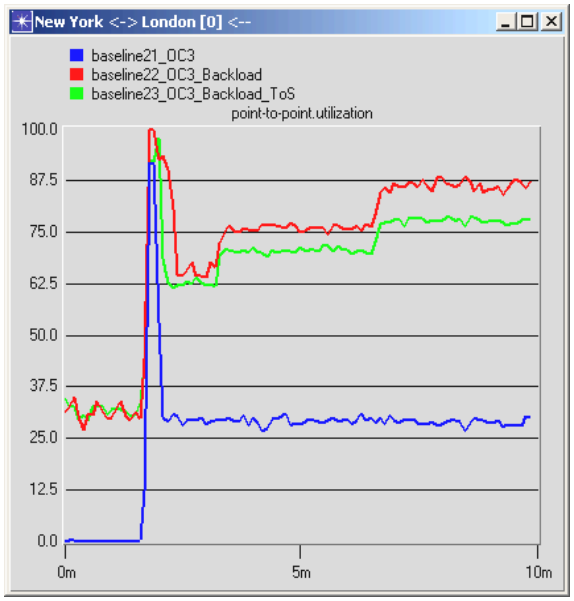


Figure 174: ICN: Utilization NY-London

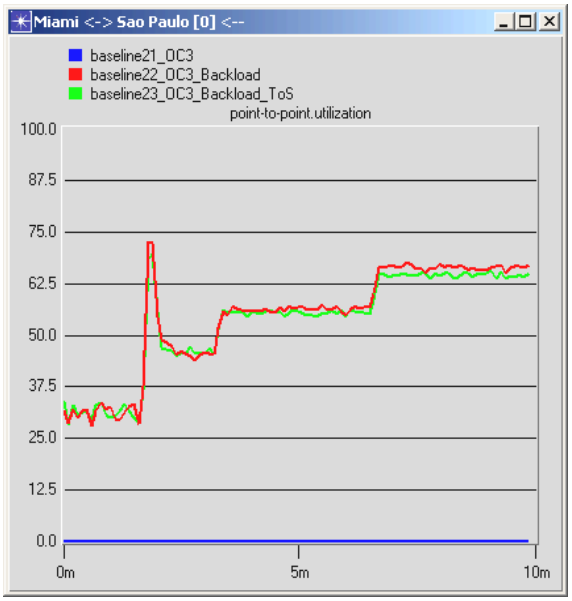


Figure 175: ICN: Utilization Miami-SP

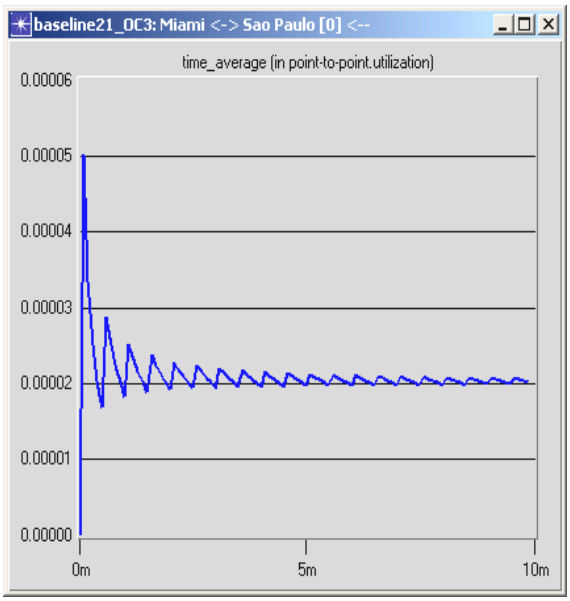


Figure 176: ICN: Utilization Miami-SP (OC3 only)

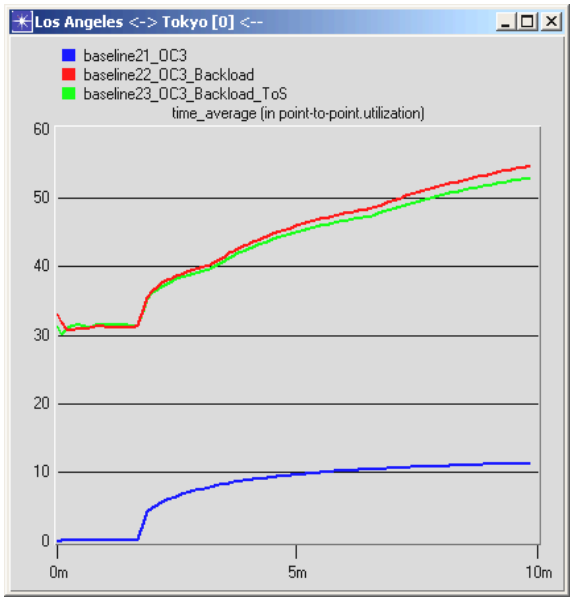


Figure 177: ICN: Utilization LA-Tokyo

. The link utilization is directly related to the throughput, i.e., it is the data traffic's throughput divided by the cable's bandwidth. OC-3 has a bandwidth of 155 Mbps – in

“techie” terms this means ‘how wide the pipe is’. Our simulations are showing some throughput values over the OC-3 bandwidth of 155 Mbps. This is due to the fact that the stated bandwidth is the “theoretical” bandwidth - the upper limit capacity “‘without degradation of the data traffic”. In reality, the cable can carry over its capacity but it is not recommended for safe and security reasons. Most of network managers recommend upgrading to a higher bandwidth if the link utilization is greater than 70% or 80%. These high levels of utilization indicate a bottleneck somewhere in the network and it can result in a rapidly increasing number of dropped packets, retransmissions, and slow response times. Utilization above 80% is possible for short periods of time, however, due to windowing and acknowledgements; the TCP protocol is unable to sustain utilization in excess of 80% for more than a few minutes at a time. On the other hand, low utilization rates cast serious doubt on the advisability of QoS measures, which are largely motivated by the assumption that networks are congested. When a link is heavily loaded, it makes sense to create a special “lane” for high priority traffic, to avoid the delays and packet losses of the bulk of the traffic. However, when a network is engineered to handle high-speed bursts that arise from mission-critical applications, it will exhibit low utilization, and all traffic will be going through expeditiously almost all the time.

. The graphs show that the link is being over-utilized even for low level of background traffic (30%). The explicit traffic utilizes an average of 22% of the link capacity for either NCN or ICN. However the utilization goes over 100% for any of the rates of background traffic, which means that OC-3 can be the link of choice to interconnect our subnets unless the background is anything above 30% of the link’s bandwidth.

11.6 ICN: Scenario 23 – Link (no T3) + Backload + QoS

This last scenario involves an analysis of ICN performance under different links and quality of service. The same was done before with NCN and in the last section with the OC-3 link. The simulations results give us an opportunity to compare these settings under the same conditions. The graphs are the results of two scenarios: one that includes the link with the explicit traffic, and the other is the link carrying the hybrid traffic (explicit plus background traffic) and QoS. The analyses are similar to the ones done so far in this thesis with a particular similarity with the previously section where OC-3 was the link used to interconnect all the subnets. The difference between these two situations is that now the analysis does not include the scenario where there is only hybrid traffic on the link without QoS, because this scenario could not be run with the computing resources at hand. The links used in this section are all optical cables: OC-3, OC-12, and OC-48. The addition of background traffic in the campus network is usually associated with extranet traffic. Many faculty, staff and students use commercial ISPs when they are working out of the campus environment; virtually all traffic from commercial ISPs passes through the campus connection to the Internet. Usually, the largest portion of a campus Internet usage is outgoing Web traffic; that is, people using the campus connection to the Internet looking at content on the campus Web servers. These people might be current faculty or students using their home DSL connections, they might be faculty or students from other academic institutions, they might be prospective students doing research on universities, or they might be members of the general public browsing the wealth of information the campus departments provide via the Web.

11.6.1 ICN: ETE delay – Link (no T3) + Backload + QoS

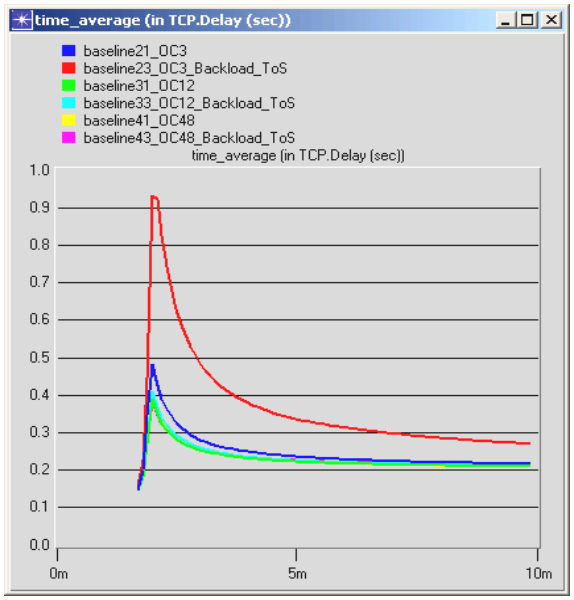


Figure 178: ICN: TCP delay

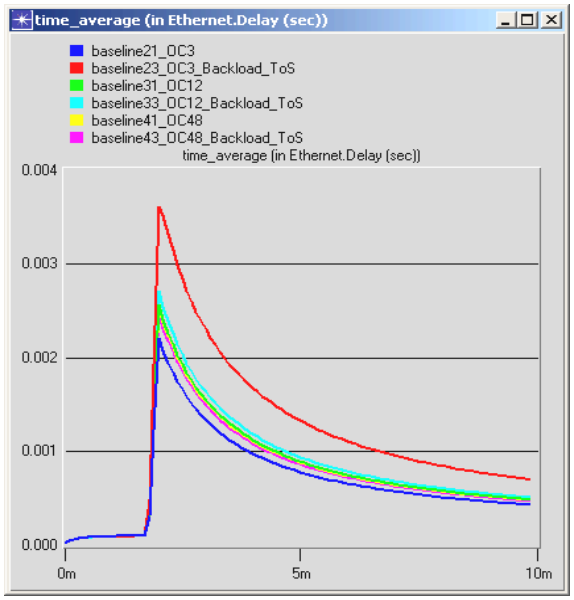


Figure 179: ICN: Ethernet delay

. TCP delay graph shows the drop-tail feature which is common to most of the previous graphs. Also, there is the convergence to a steady value (210 ms) occurring towards the end of the simulation run for the faster cables. The presence of QoS does not alter the delay for OC-12 and OC-48), but it increases the delay for OC-3 from 220 ms to 270 ms. The converged value of 210 ms is twice the value for NCN, and it is thus compatible with the fact that NCN has half of the traffic of ICN.

. Ethernet delay reproduces the features of TCP in a smaller scale, in microseconds. The converged value is 480 microseconds, which is twice the value for NCN.

11.6.2 ICN: Response Times – Link (no T3) + Backload + QoS

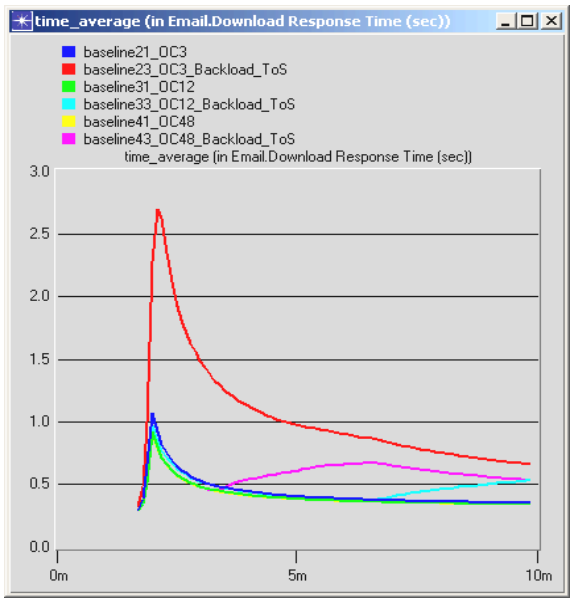


Figure 180: ICN: E-Mail

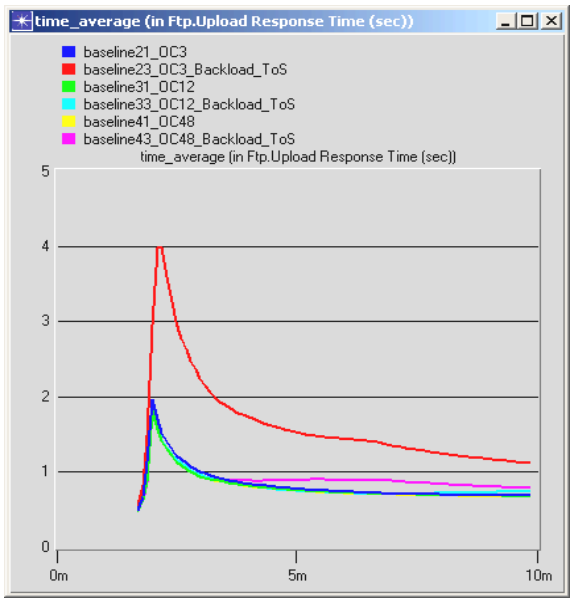


Figure 181: ICN: FTP

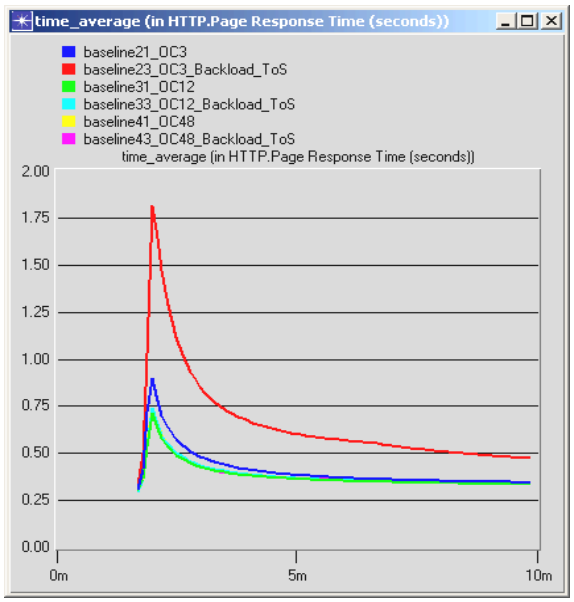


Figure 182: ICN: HTTP

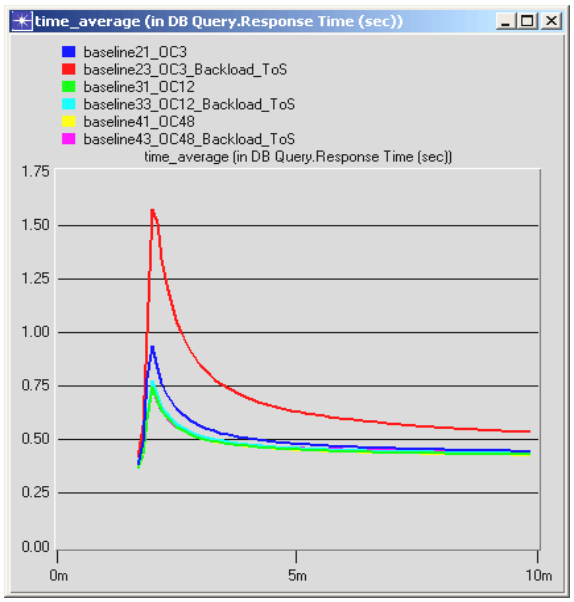


Figure 183: ICN: DB Query

. E-Mail: All of the response times for the links with only explicit traffic converge to 0.35 seconds. When QoS is introduced together with background traffic things start to change. OC-3 email response time goes up from 0.35 sec to 0.66 sec. OC-12 stays at 0.35 sec until when the background level goes up 50% of link capacity then the response reaches 0.53 sec. When the background traffic level is up to 40% of the OC-48 link capacity, the response time for e-mails moves up to 0.67 sec and when the background level is 50% the response time goes down to 0.53 sec – the same ending value as for OC-12. These changes reflect the sensitivity of response times towards background traffic and QoS.

. Note that the background traffic increases the throughput and consequently increases the response times for the applications. In turn, QoS mechanism reduces this increase in these response times by better management of the routers' queues. This is one the central point of thesis which suggests QoS in the network as a method to reduce delay and response times – indeed these are the main goals of user and network managers.

. FTP: As with E-mail response times, the response times for FTP are the same for the undisturbed links with a steady value of 0.68 seconds. The background traffic and QoS change this constant value: for OC-3 the response time ends up at 1.12 seconds; for OC-12 the graphs shows an increase to 0.73 sec when the backload is 50%; and for OC-48 the response time goes up to 0.89 sec when the backload is 30% (200 sec of simulation time) and then it decreases to 0.78 sec when the backload is 50% of the link bandwidth.

. HTTP: There is no change whatsoever for Web page response time for any of the links – they all converge to 0.35 seconds, even when QoS is present, with exception for OC-3 which shows an increase from the converged value of 0.35 seconds to 0.48 seconds when QoS is included in the network.

. DB Query: The behavior of response times for database queries is similar to that for HTTP. The response times for undisturbed links OC-12 and OC-48 and for the links with backload and QoS converge to a value - 0.43 seconds. While for OC-3, the backload with QoS increases that value to 0.54 seconds. It is important to know that DB Query has the largest traffic of all the applications, over 300% higher than the second largest application, FTP; and still, even with this amount of traffic plus the background traffic, QoS managed to reduce by 20% the response time.

. Recall that the analysis done of the three scenarios for OC-3: link, link + backload, and link + backload + QoS, revealed that the backload results in longer response time than the undisturbed link, and that QoS shorten that increase specifically for database queries which are traffic with higher priority.

11.6.3 ICN: Queue delay – Link (no T3) + Backload + QoS

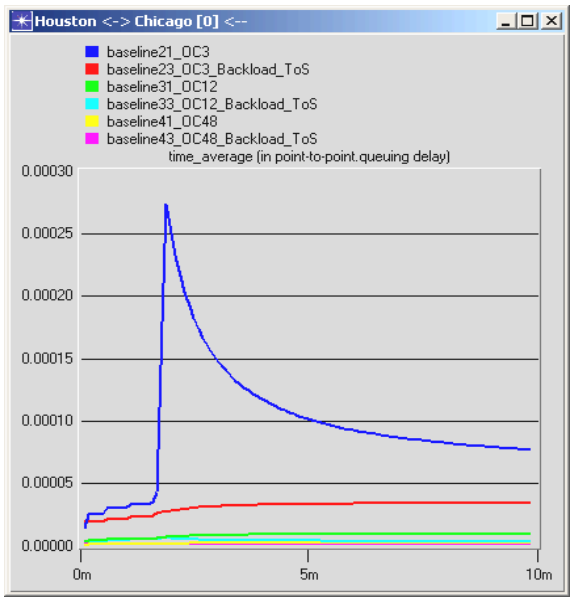


Figure 184: ICN: Queue delay Hou-Chi

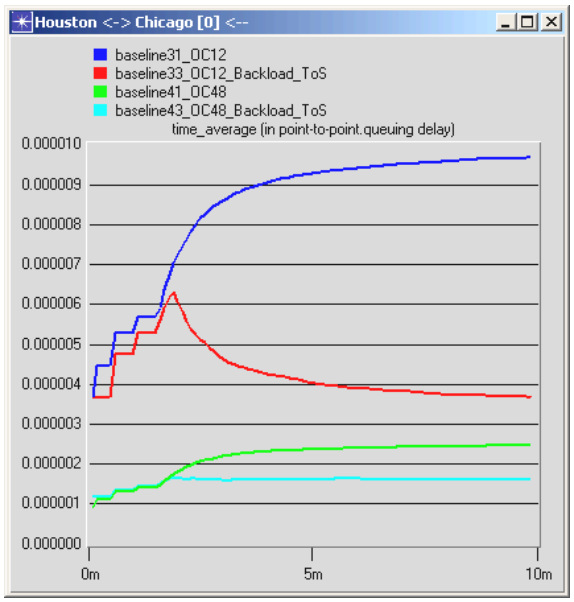


Figure 185: ICN: Queue delay Hou-Chi no OC3

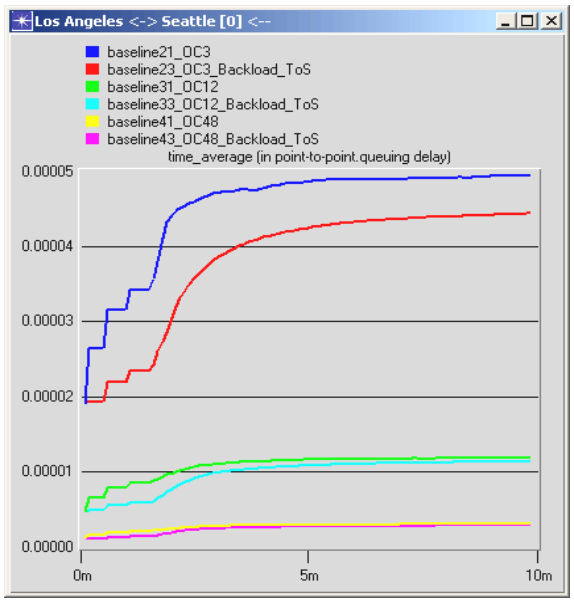


Figure 186: ICN: Queue delay LA-Seattle

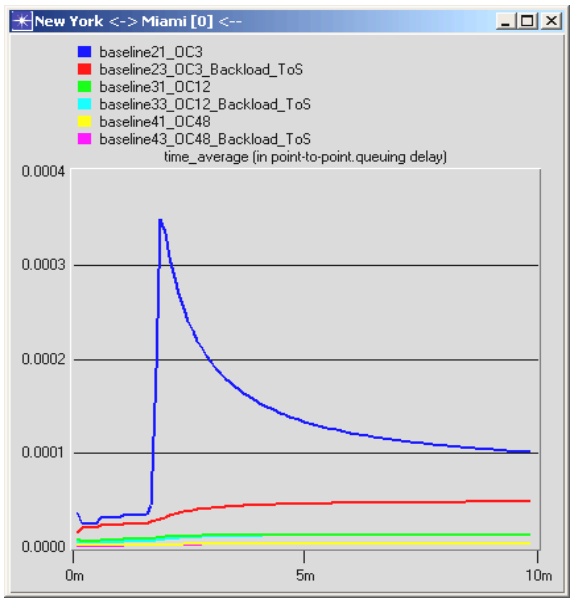


Figure 187: ICN: Queue delay NY-Miami

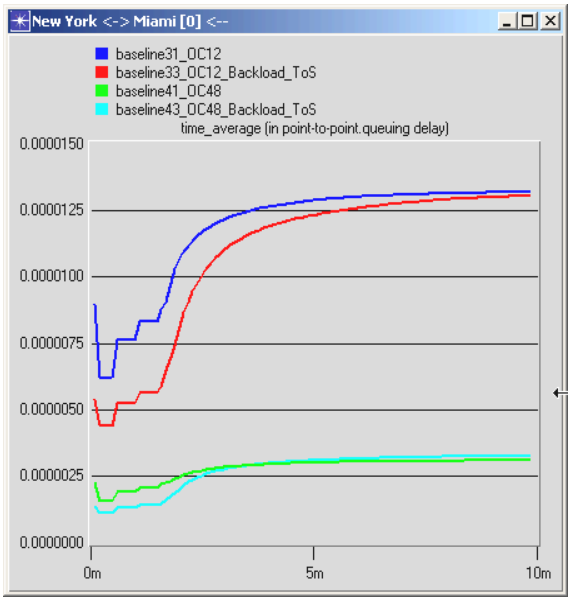


Figure 188: ICN: Q delay NY-Miami no OC3

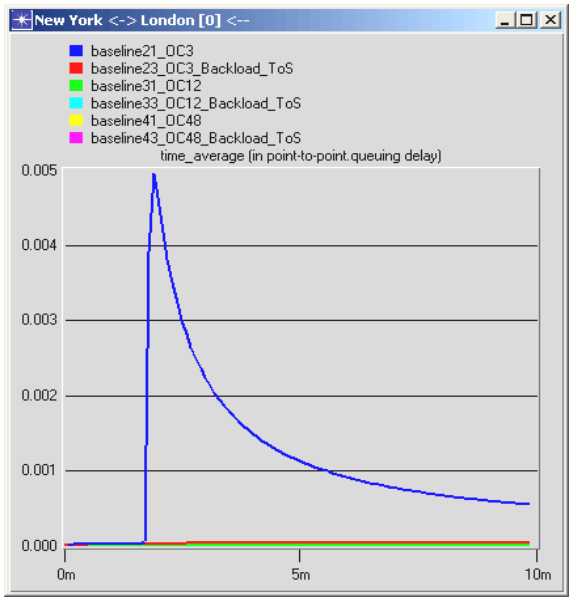


Figure 189: ICN: Q delay NY-London

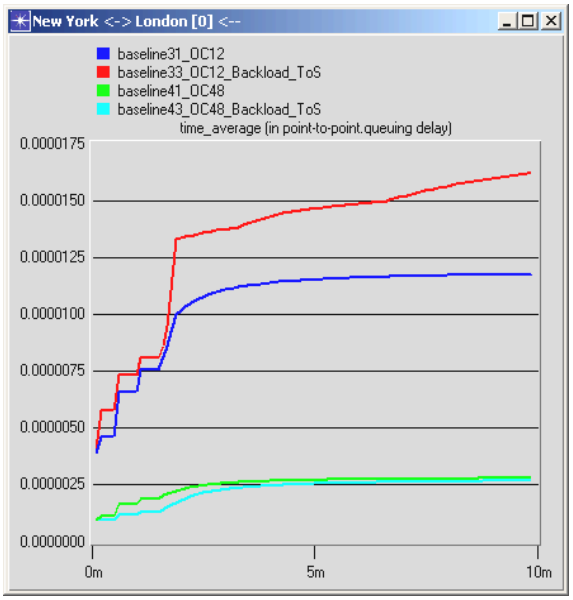


Figure 190: ICN: Queue delay NY-London no OC3

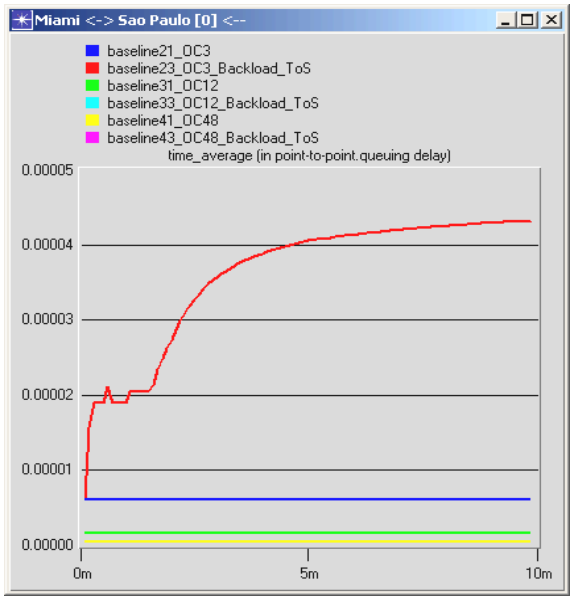


Figure 191: ICN: Queue delay Miami-SP

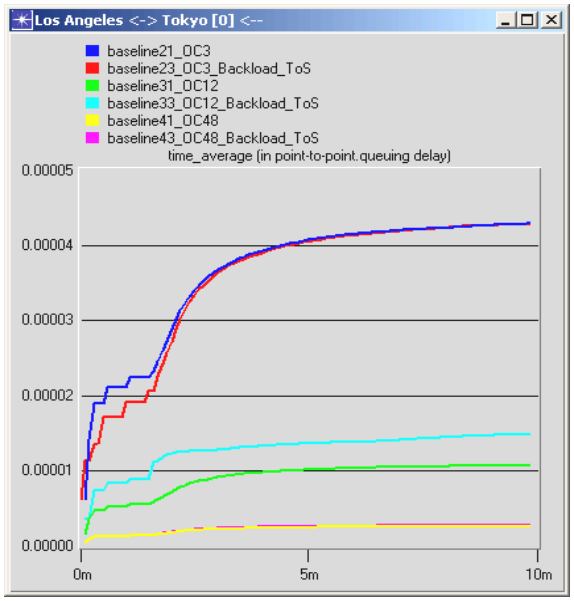


Figure 192: ICN: Queue delay LA-Tokyo

. The graphs for LA-Seattle, NY-Miami, and Miami-Sao Paulo show practically no changes in queue delay when comparing link (either OC-12 or OC-48) carrying explicit traffic with the same link carrying hybrid traffic with QoS. For these scenarios, the queue delay in each of the subnets of these pairs is insensitive to QoS. For Houston-Chicago, the mid-way link of the entire network, the presence of QoS resulted in the expected reduction in the queue delay for OC-12 and OC-48.

However, for the subnet pairs of NY-London and LA-Tokyo connected via OC-12 link, the queue delay actually increases with QoS, although infinitesimally. The traffic is being trapped longer in the queues of these subnets. LA-Tokyo queue delay is not influenced by higher levels of background traffic, while the NY-London link shows the queue delay line going upward when 50% of the link bandwidth is occupied by the background traffic. Distance could be a factor, however the routers are using RIP as the routing

engine and therefore only hop count is a factor in deciding the route. The most plausible cause for this behavior is the fact that LA and NY are the subnets that connect the national network with the international subnets are points of large flow of traffic and therefore the packets in the queues are taking longer to leave the queues and be transmitted over the link. This situation is back to normal, due to the efficiency of WFQ - the queuing mechanism implemented in the routers, for the nearest subnets to NY and LA. Thus, for example in the Miami-Sao Paulo link, which is also connecting a domestic subnet with an international subnet, the queue delay is unchanged with QoS.

Analysis for OC-3 was already done in the last section and it showed the reduction in the queue delay caused by QoS in all the subnet pairs.

. Table 22 shows the queue delay (in microseconds) for the subnet pairs (only the first letter is used to denote the subnet name) and the links (L), background (B), and QoS (Q).

	H-C		**L-S**		**N-M**		**N-L**		**M-S**		**L-T**	
	L	+B+Q	L	+B+Q	L	+B+Q	L	+B+Q	L	+B+Q	L	+B+Q
OC3	130	89	110	93	130	93	180	93	8	94	100	95
OC12	26	16	25	24	25	25	26	31	2	2	24	27
OC48	7	6	6	6	6	6	6	6	1	1	6	5

Table 22: ICN Subnet Pairs Queue Delay (microseconds)

11.6.4 ICN: Throughput – Link (no T3) + Backload + QoS

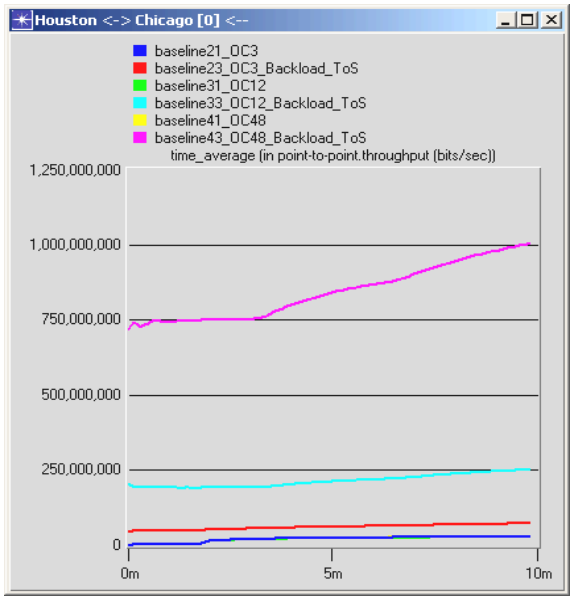


Figure 193: ICN: Throughput Hou-Chi

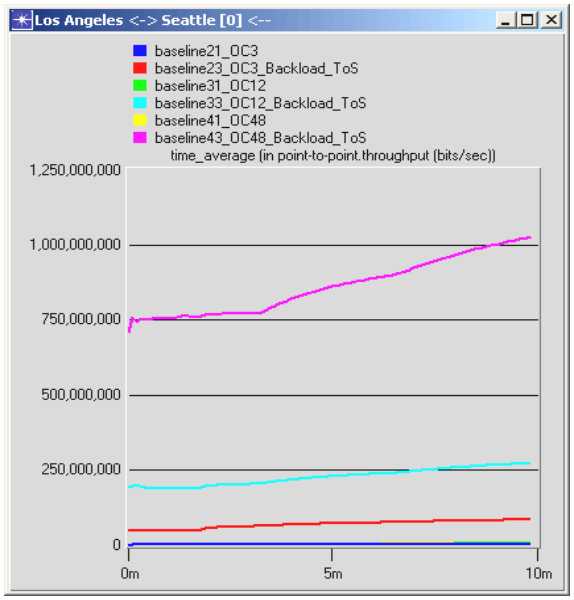


Figure 194: ICN: Throughput LA-Seattle

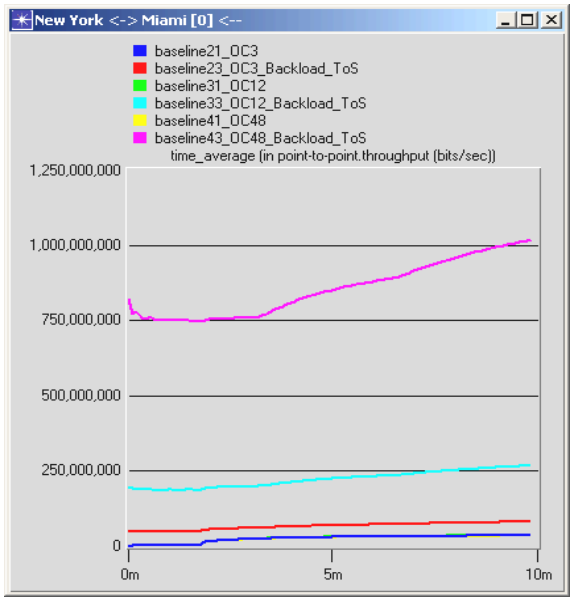


Figure 195: ICN: Throughput NY-Miami

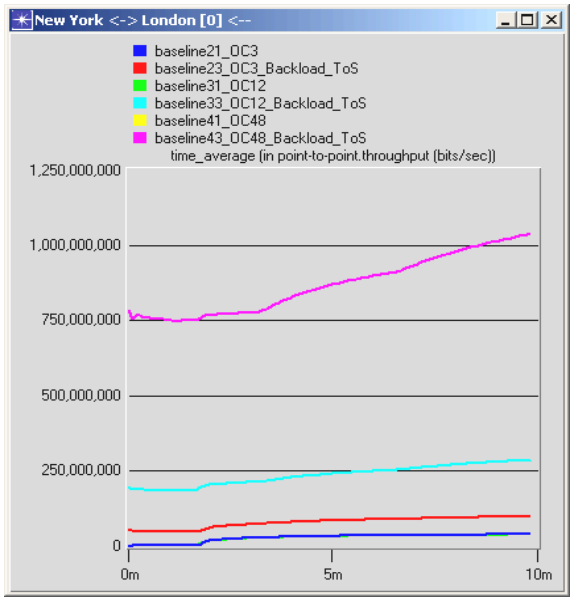


Figure 196: ICN: Throughput NY-London

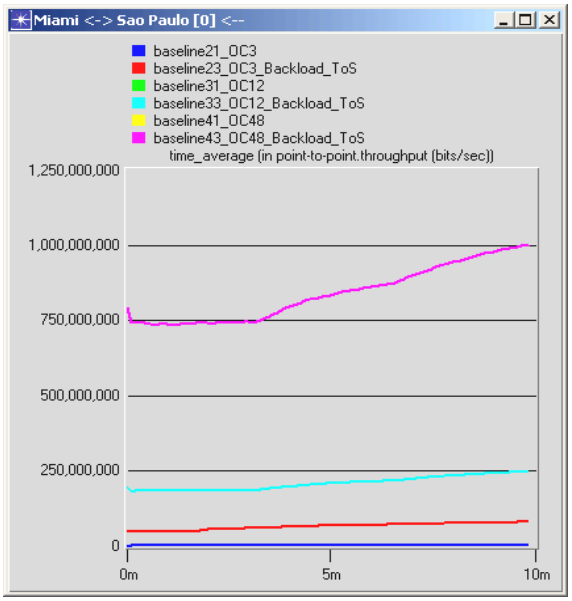


Figure 197: ICN: Throughput Miami-SP

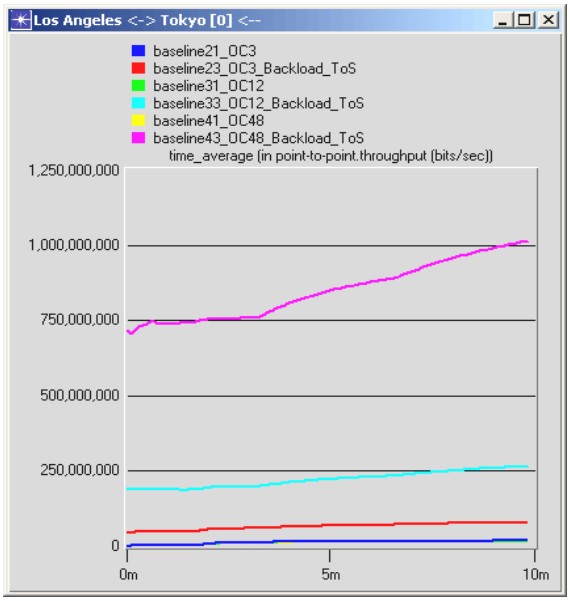


Figure 198: ICN: Throughput LA-Tokyo

. The graphs above show throughput for one direction only. Therefore, one needs to double the values to get the total throughput. However, due to various inefficiencies of the simulator implementation, close analysis was done for every single subnet pairs and for every single link used to get the total throughput.

. Throughput is independent of the bandwidth of the link, except when the bandwidth of the link is smaller than the amount of traffic flowing on it. In this case, the throughput causes congestion and ultimately might put at risk devices and users.

. The average throughput (Mbps) for the subnet pairs for the links only carrying explicit traffic is constant, range from 89 Mbps for NY-London, to 60Kbps for Miami-Sao Paulo.

. The optical cables are built in such a way that their bandwidths increase by around 300% from the lower denomination cable to higher denomination one. Thus, OC-12 bandwidth is 300% higher than the bandwidth of the OC-3 link, and the bandwidth of OC-48 is around 300% of that of OC-12.

It is an interesting finding that for each of the subnet pairs with the hybrid traffic and QoS present, the percentage increase of the throughput is around 300% from OC-3 to OC-12, and from OC-12 to OC-48. This is in direct relation to the relationship between these cables' bandwidths.

. The throughputs (Mbps) for the subnet pairs and the links used are shown in table 23.

	H-C		**L-S**		**N-M**		**N-L**		**M-S**		**L-T**	
	L	+B+Q	L	+B+Q	L	+B+Q	L	+B+Q	L	+B+Q	L	+B+Q
OC3	29	143	22	172	43	157	81	199	0.06	159	35	159
OC12	32	506	20	545	50	528	93	578	0.06	497	41	533
OC48	33	2009	21	2054	50	2027	93	2084	0.06	2003	42	2035

Table 23: ICN Subnet Pairs Throughput (Mbps)

11.6.5 ICN: Utilization – Link (no T3) + Backload + QoS

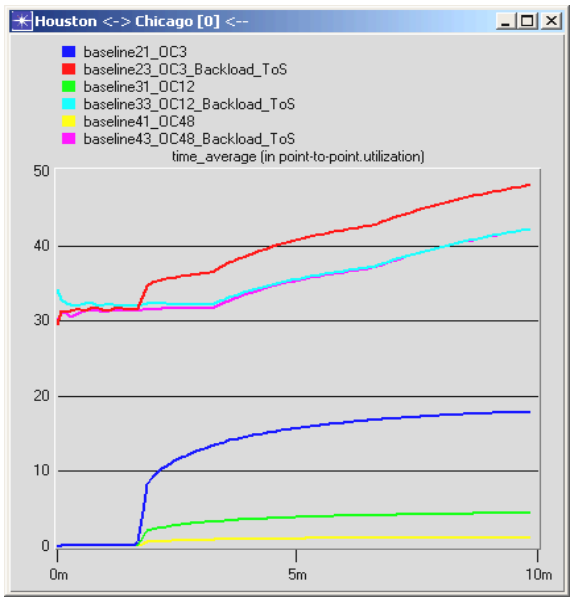


Figure 199: ICN: Utilization Hou-Chi

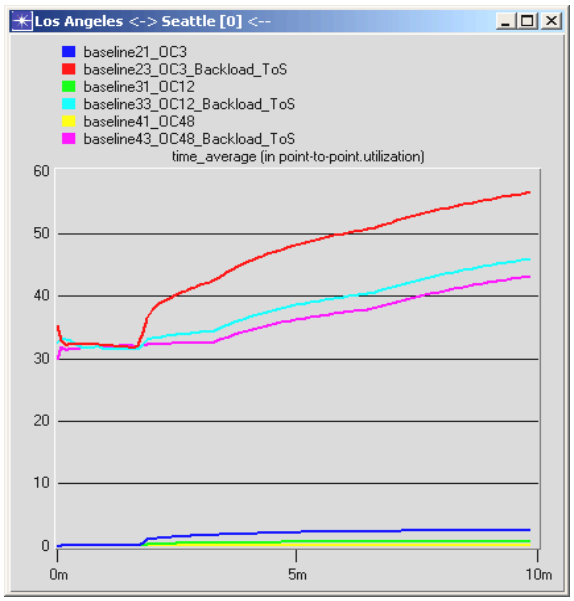


Figure 200: ICN: Utilization LA-Seattle

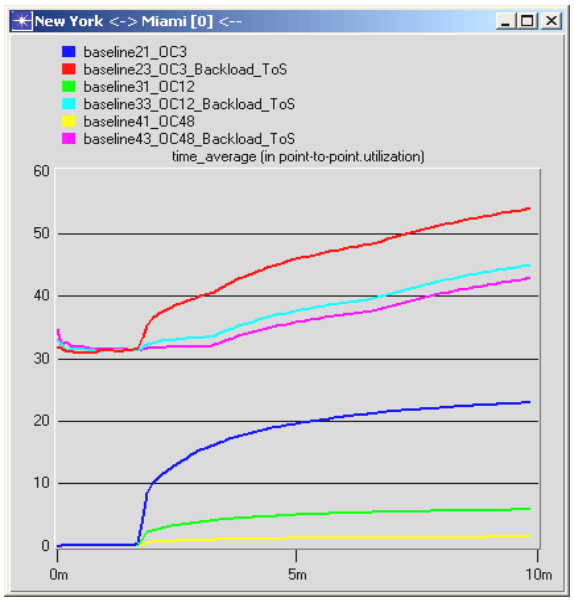


Figure 201: ICN: Utilization NY-Miami

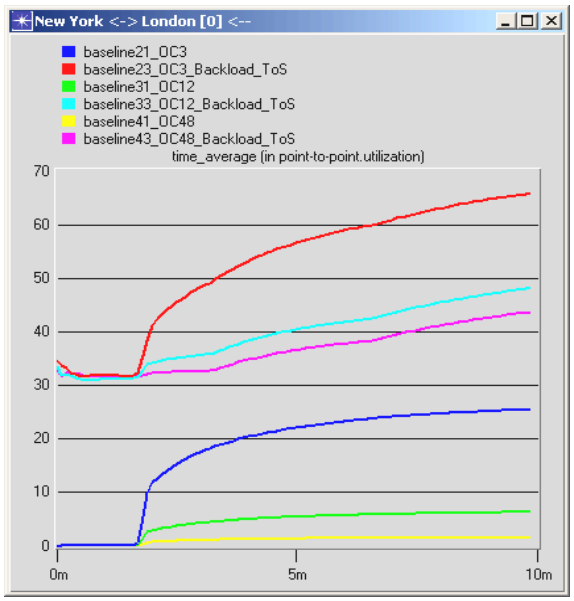


Figure 202: ICN: Utilization NY-London

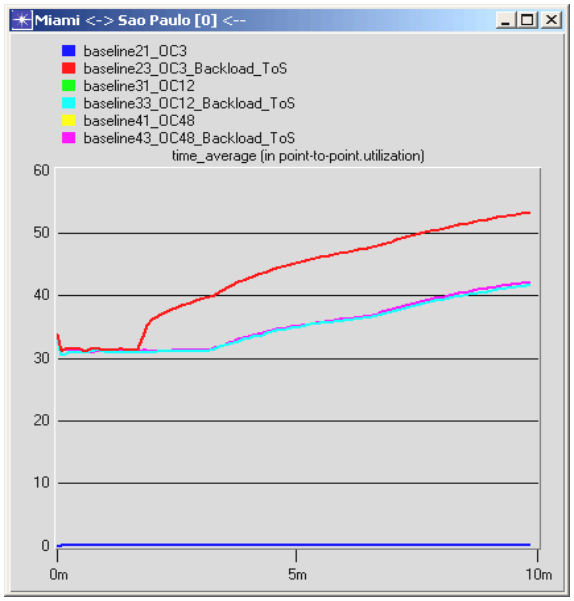


Figure 203: ICN: Utilization Miami-SP

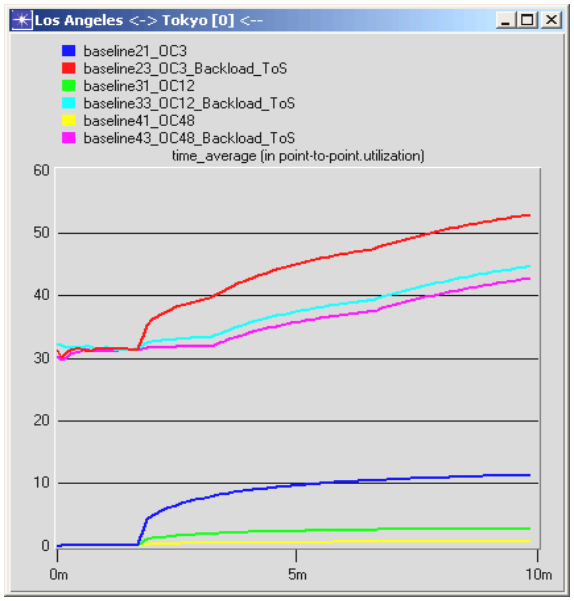


Figure 204: ICN: Utilization LA-Tokyo

. The graphs shown above represent link utilization for one direction only. The final statistics used in the analysis that follows were the results of both graphs, one in each direction. For some of the subnet pairs the total utilization could be obtained by just multiplying by 2 the mean value of utilization in one of directions, however, for other subnet pairs the final numbers were different from the numbers resulting from doubling the individual statistics. The reason for this is assumed to be the various inefficiencies of the simulator implementation.

. From the view point of long-term network growth, OC-3 would not be the ideal link for ICN, since utilization for undisturbed link is over fifty percent for the subnet pair with the largest amount traffic, NY-London. Also, NY-Miami shows 30% utilization of the OC-3

link. Either OC-12 or OC-48 can be the ideal link for the ICN; with these optical cables and the amount of traffic in the network, utilization is shown to be in the teens.

. Regarding the background traffic and QoS, none of the optical cables support the network traffic with the degradation and type of services (ToS) levels adopted in this research. Filtering switches can be used to isolate backloads and lower the utilization for this type of extraneous traffic.

. Table 24 summarizes the link utilization (%) for the subnet pairs and links.

	H-C		**L-S**		**N-M**		**N-L**		**M-S**		**L-T**	
	L	+B+Q	L	+B+Q	L	+B+Q	L	+B+Q	L	+B+Q	L	+B+Q
OC3	19	96	11	116	29	106	54	134	0.05	107	24	107
OC12	5	85	4	92	8	89	16	97	0.01	84	7	90
OC48	1	85	1	86	2	85	4	88	0.003	84	2	86

Table 24: ICN Subnet Pairs Link Utilization (%)

11.7 Real Network Traffic

This thesis has Abilene, the Internet2 network, looming in the background as its source of information and inspiration. The statistics of real subnet pairs that are part of Abilene is shown below for comparison with some of the thesis results. The difficulty of getting information from real campus networks feels like insurmountable sometimes such is number the layers of authorities that one has to go through in order to have access to a piece of information. Network administrators are very protective of their product and for good reasons. But then this becomes an obstacle for anyone to inspect or share information about the network. In addition, besides "routing the network traffic around the campus", as one of them put it, not much else information are being gathered in the campus gateways. One can understand the reason for it, any tick of the resources time should be devoted only and exclusively to "route the network traffic around the campus". However, much information could save time and money if the resources could also be set up to collect these network parameters that can be used to optimize the network's performance, such as delays an response times.

Hence, this first part of the thesis can be used as a groundbreaking material for serious work on campus network performance. The objective can either be how to maximize the existing resources by including traffic differentiation or how to optimize performance through the upgrade of the network links and interfaces.

The graphs of actual Internet 2 traffic shown below for comparison reasons with our results were taken straight from several of the Abilene's peer networks Web sites.

11.7.1 Actual Throughput – Abilene Peers

. New York – London

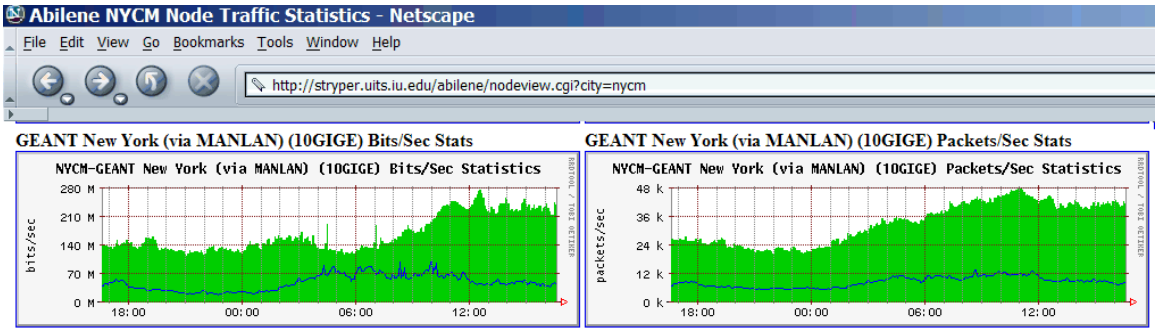


Figure 205: Actual Traffic NY-London (Mbps)

The peak out-traffic of 70 Mbps compares well with the figure 135.

. Miami – São Paulo

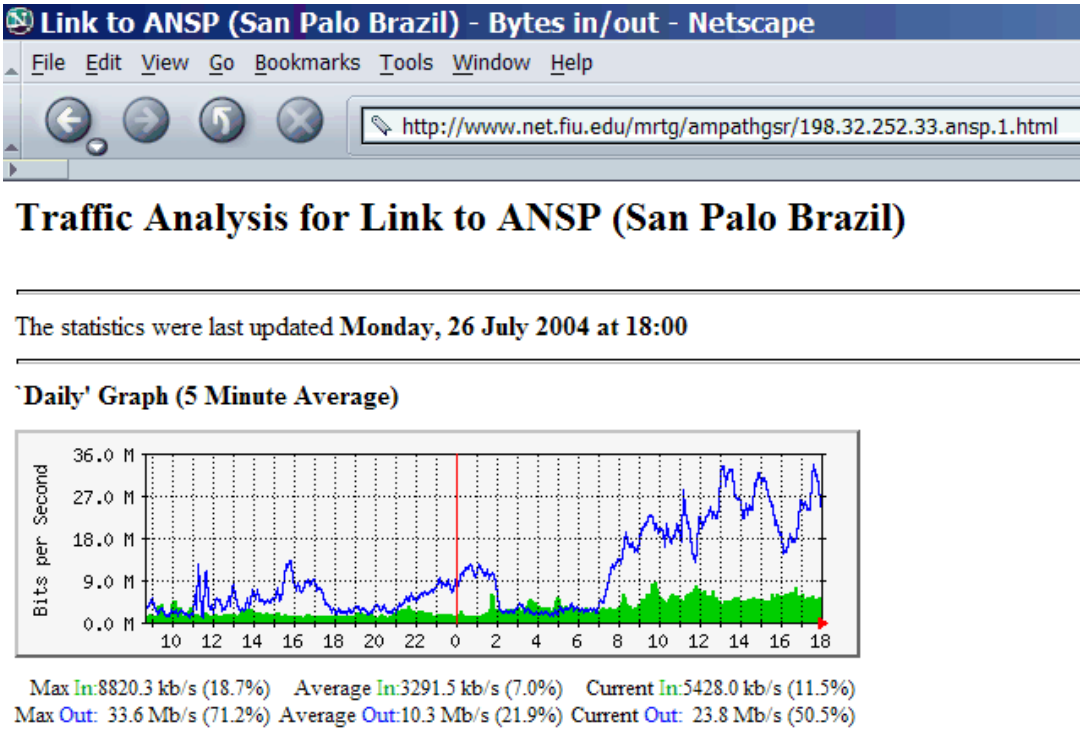


Figure 206: Actual Traffic – Miami-Sao Paulo

Actually, Abilene runs a T-3 line connecting Miami to São Paulo.

Figure 207 shows the simulation graph of throughput of this subnet pair linked by a T-3 line.

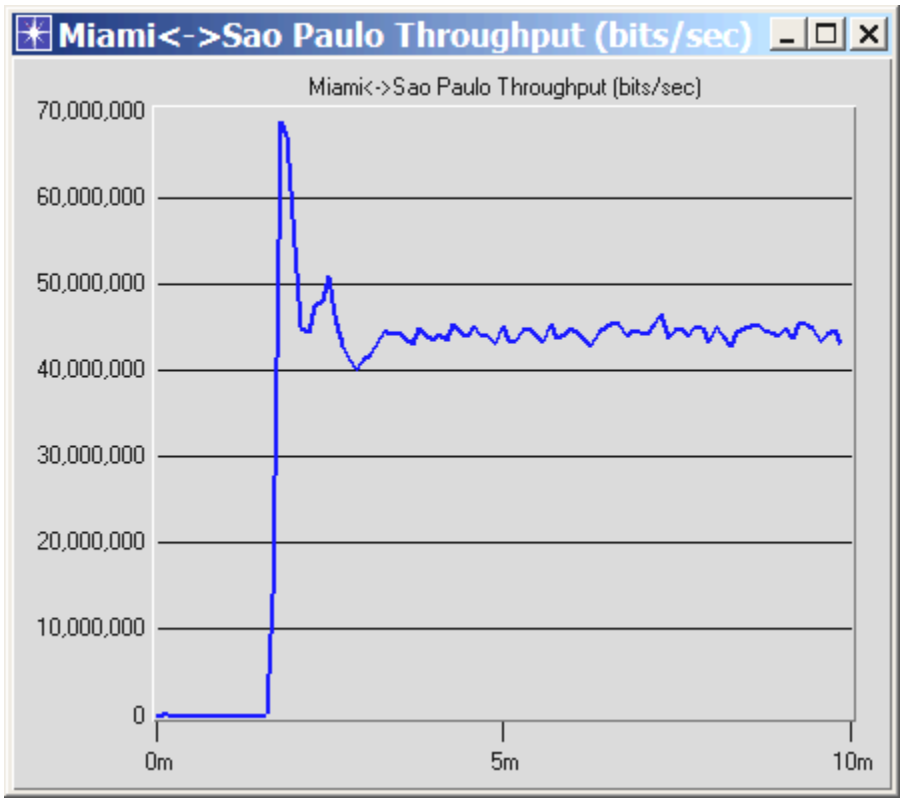


Figure 207: Actual Traffic Miami-Sao Paulo (Mbps)

Figure 207 shows that the throughput in the steady state is around 43 Mbps which is close to the real throughput at the peak time for the total traffic: outbound and inbound traffic shown in figure 206.

. Los Angeles – Tokyo

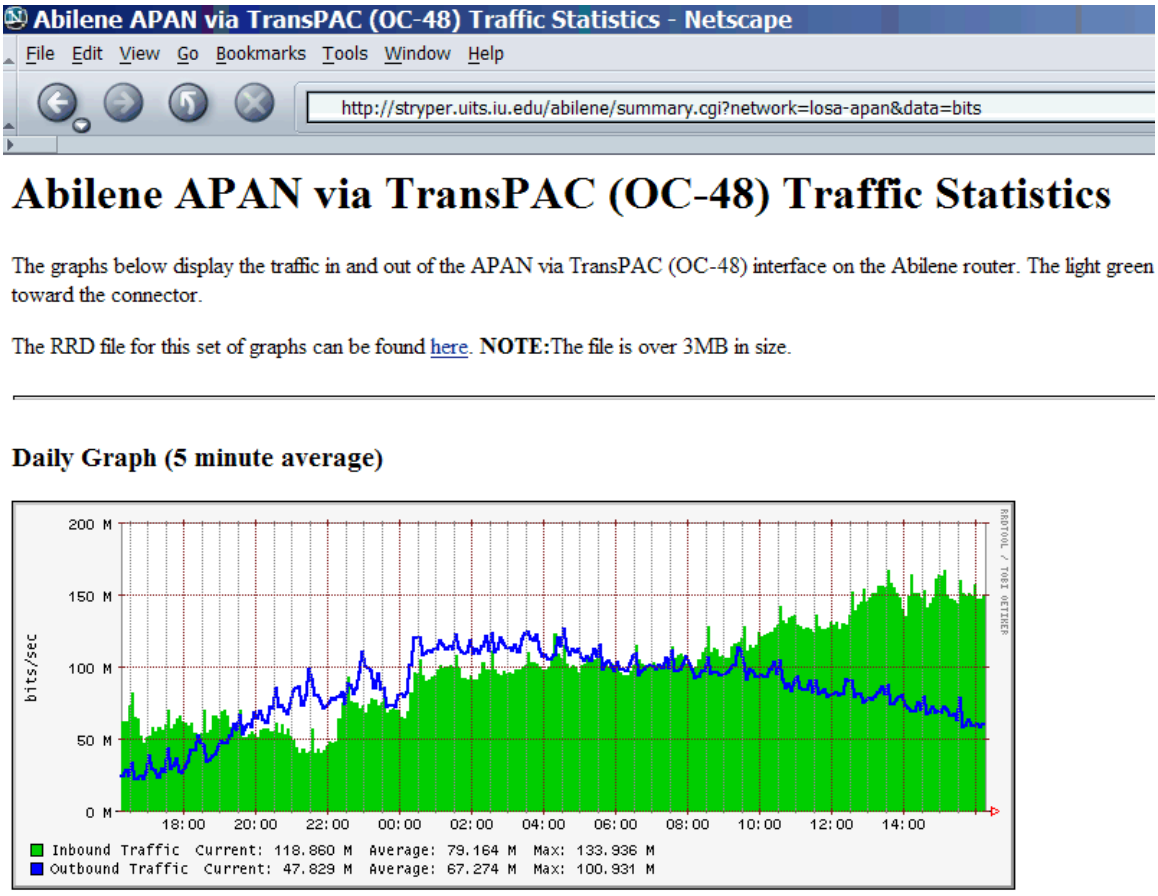


Figure 208: Actual Traffic – LA-Tokyo

Compare this with figure 169 when background is included with the explicit traffic. The throughput reaches close to 150 Mbps which is close to actual peak number of figure 208. However, figure 169 is for OC-3 link. The simulation graph for OC-48, figure 138, gives us one-direction throughput, thus if we double it we get the total throughput which is equal to 30 Mbps. Again, the figure 208 includes all of the campus networks in the Tokyo area which with certainty is larger than our 1,000-user campus.

. New York – Chicago

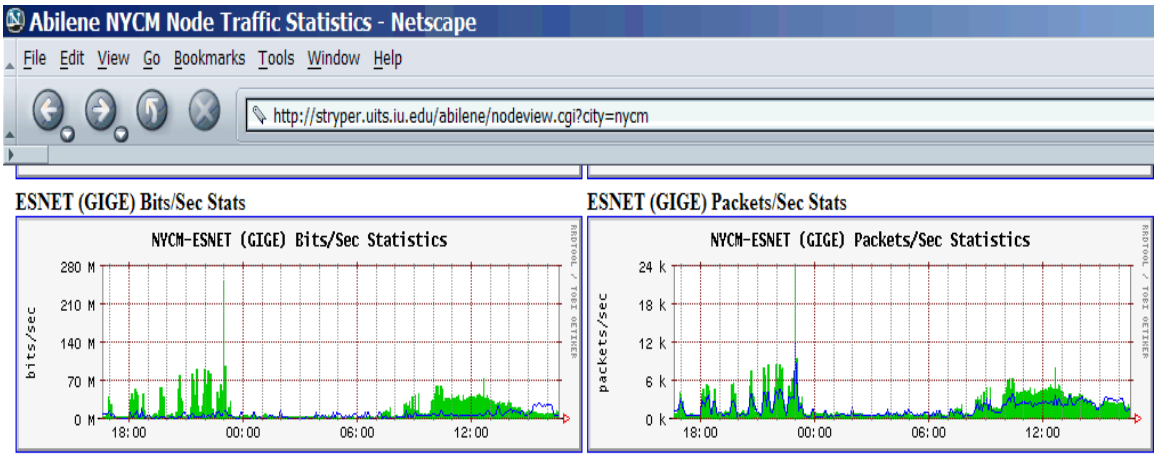


Figure 209: Actual Traffic – NY-Chicago (Mbps)

The simulation throughput shown in figure 210 is close to the actual shown in figure 209.

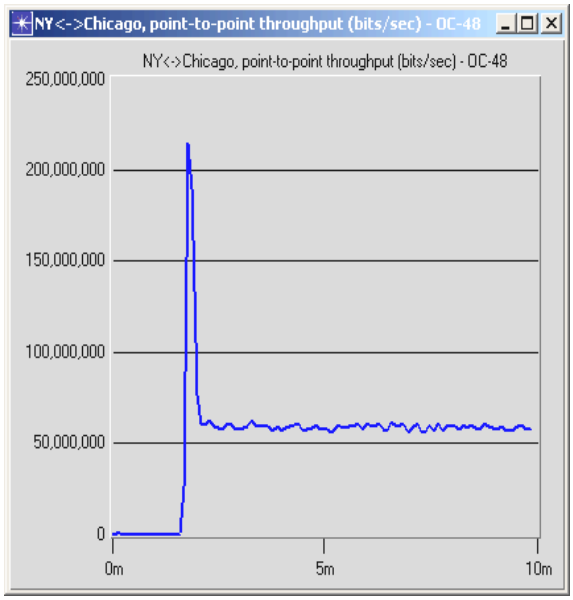


Figure 210: Throughput NY-Chicago

Figure 209 includes all of the campuses located in the New York and Chicago areas and their sizes are certainly larger than our 1,000-users, however, our "Heavy-user Ethernet Client" generates a comparable traffic to the actual one shown in figure 209.

11.7.2 Actual Round-Trip Time – Abilene Peers

. Miami – São Paulo

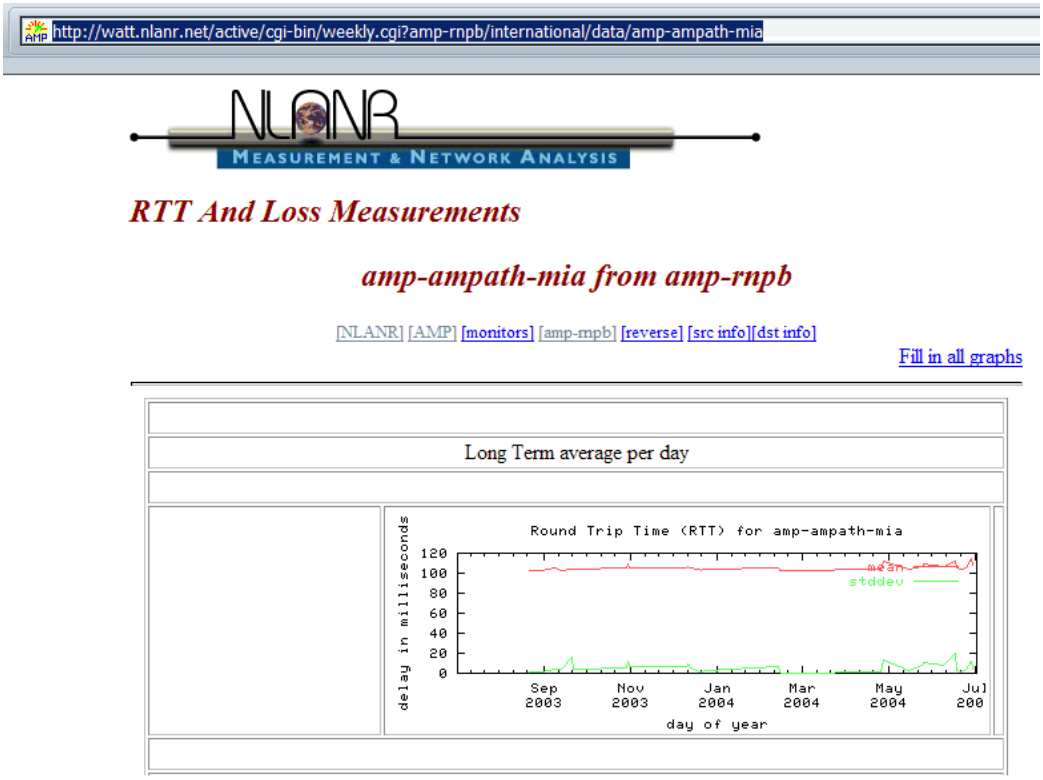


Figure 211: Actual RTT – Miami – Sao Paulo

End-to-end delay or latency is usually associated with the time it takes a message to travel from one end of a network to the other. There are many situations in which it is more important to know how long it takes to send a message from one end of a network to the other and back, rather than the one-way latency. This is called the round-trip time (RTT) of the network. The simulations did not collect information on RTT, however response times include TCP SYN-ACK handshakes which doubles the round-trip time. Our response times is around 0.25 seconds or 250 milliseconds which is in the same order of magnitude as shown in figure 211.

12. Intelligence in Networks

Networks provide us with the opportunity to link our operations through giga-stream channels, nationally as well as internationally, as if they were in the same geographical place, and at the same time ensure 24-hour servicing to any user.

Network intelligence requires more than a set of connected elements; it requires an interconnecting and functionally coupled architecture that enables the various functional levels to interact and communicate efficiently with each other. Network intelligence will evolve through growth in computational power and through the accumulation of knowledge about the types of input data needed for making decisions concerning expected response, and about the algorithmic processing required in complex and changing communications environments. Increasingly sophisticated network intelligence makes possible look-ahead planning, management before responding, and reasoning about the probable results of alternative actions. These network capabilities can provide service providers with competitive and operational advantages over traditional networks.

The degree or level of network intelligence is determined by: (i) the computational power of the network and the network elements, (ii) the sophistication of the algorithms the network system uses, and (iii) the information and data values the system can store and access. More sophisticated levels of network intelligence include the ability to recognize user, application, service provider, and infrastructure needs, as well as expected and unexpected events, the ability to represent knowledge in a world model, and the ability to reason about and plan for the future.

13. Intelligent Network (IN)

The term ‘intelligent network’ was coined in 1986 by Bellcore, the R&D laboratory of the former AT&T operating companies, and it has become quite popular. Unfortunately, Bellcore only described the intelligent network concept – it did not define it. As a result, practically every telco today gives its own definition and interpretation.
Intelligent networks were first developed to address the problem of message switching for long-distance telephone networks (toll networks). One of the main goals of the original concept of Intelligent Networks (IN) was to reduce the time needed for introducing new telephony services.

The telephony service is very simple. Its objective is the transport of speech information over a distance in real time. Telephony networks were originally designed with the assumption that the same service would be offered to all users, and this held true for a long time. Users could select one of a range of destinations and be called by other users. Over the years the focus of telephony service providers has been to improve the technology to offer these basic services to a larger number of customers, and over longer and longer distances. At the same time, terminals have become mobile, with mobile phone users demanding the same levels of services.

In this context, the task of provisioning a new service in an operator's network is extremely complex and may vary considerably depending on the network infrastructure. This situation has led network operators and their suppliers to look for a better approach, in which control functions and data management linked to the creation, deployment, and

modification of services can evolve separately from the basic switching or existing functions of an exchange. They assigned standards organizations (ITU-T, ETSI, BellCore) the responsibility of defining an architectural framework for the creation, execution, and management of network services.

The first practical uses of the IN concept were the software-controlled switches in the 1970s, which were used in telephone networks that connected very few telephone sets. These switching systems had begun to be deployed in the local service networks providing line-oriented services such as Call Waiting and Call Forwarding. However, the toll network at this time basically had the function of moving large amount of telephone traffic efficiently between metropolitan areas. The toll network backbone, No. 4A Crossbar switch, largely implemented simple digit manipulation functions (translating, deleting, and prefixing digits in blocks of three or six) to perform the necessary routing operations. In 1976, an all-electronic switch, 4ESS, was introduced and with it a few value-added features such as the toll-free service through telephone numbers beginning with the prefix 800. At this point, the signaling network CCIS (Common Channel Interoffice Signaling) played a key role to improve efficiency and to become the basis for new network services.

The next step was to introduce a network database, known as the Network Control Point (NCP), connected to the CCIS network so that, with appropriate addressing, it could be interrogated by any switch that was also in the CCIS network. This work culminated in what was known as the Direct Services Dialing Capabilities (DSDC) architecture.

The architecture thus consisted of three basic components: NCPs, a set of switch functions for access to DSDC, also known as Action Control Point (ACP), a set of service-independent commands with associated parameters which could be exchanged between the ACP and NCP to realize different services, and Intelligent Peripheral (IP), the first instance of which was called the Network Services Complex (NSCX) and which supported features requiring touch-tone and voice prompt interaction with callers. The first services implemented using DSDC were Advanced 800 and Software-Defined Network (SDN). SDN was the first instance of another key IN-supported service: Virtual Private Network (VPN), which allows customers to economically use and manage a portion of the public network for their private network traffic. New developments are being done in providing VoIP (voice over IP) carried by traditional telecommunications carriers.

13.1 INCM Model

Any telecommunication network that provides the exact level of service the user desires is called an Intelligent Network which has as its goal to reduce the time it takes to implement these customer's services. The models for such networks suggested by the international standardization organizations, such as ITU-T (International Telecommunication Union – Telecommunication) and IEEE (Institute of Electric and Electronic Engineering), rely on the basic concept of layering which was implemented in the existing data telecommunication networks, public switched telephone networks

(PSTNs) and later in narrowband and broadband in the integrated services digital network (ISDN).

A stripped down model of IN is centered around the centralized service control located in SCPs - Service Control Points which are network nodes that contain the service logic that controls the IN-based services [9].

The first standardized model for the intelligent network, INCM, Intelligent Network Conceptual Model, is used as a tool in the development of what are called IN capability sets (CS).

The model is organized around three principal elements:

(a) call processing – operation of setting up and disconnecting a call,

(b) "hooks" – an API (Application Programming Interface) capable of starting, completing, suspending, and resuming IN operations, and

(c) IN service logic – software for supplementary IN services.

INCM consists of a four-plane structure: (i). Service plane, (ii) Global Functional Plane (GFP), (iii) Distributed Functional Plane (DFP), and (iv). Physical Plane .

Figure 212 shows all the planes of the IN model with their respective functionalities which take place during any service feature, SF. Note that each service feature is implemented through Global and Distributed service logic.

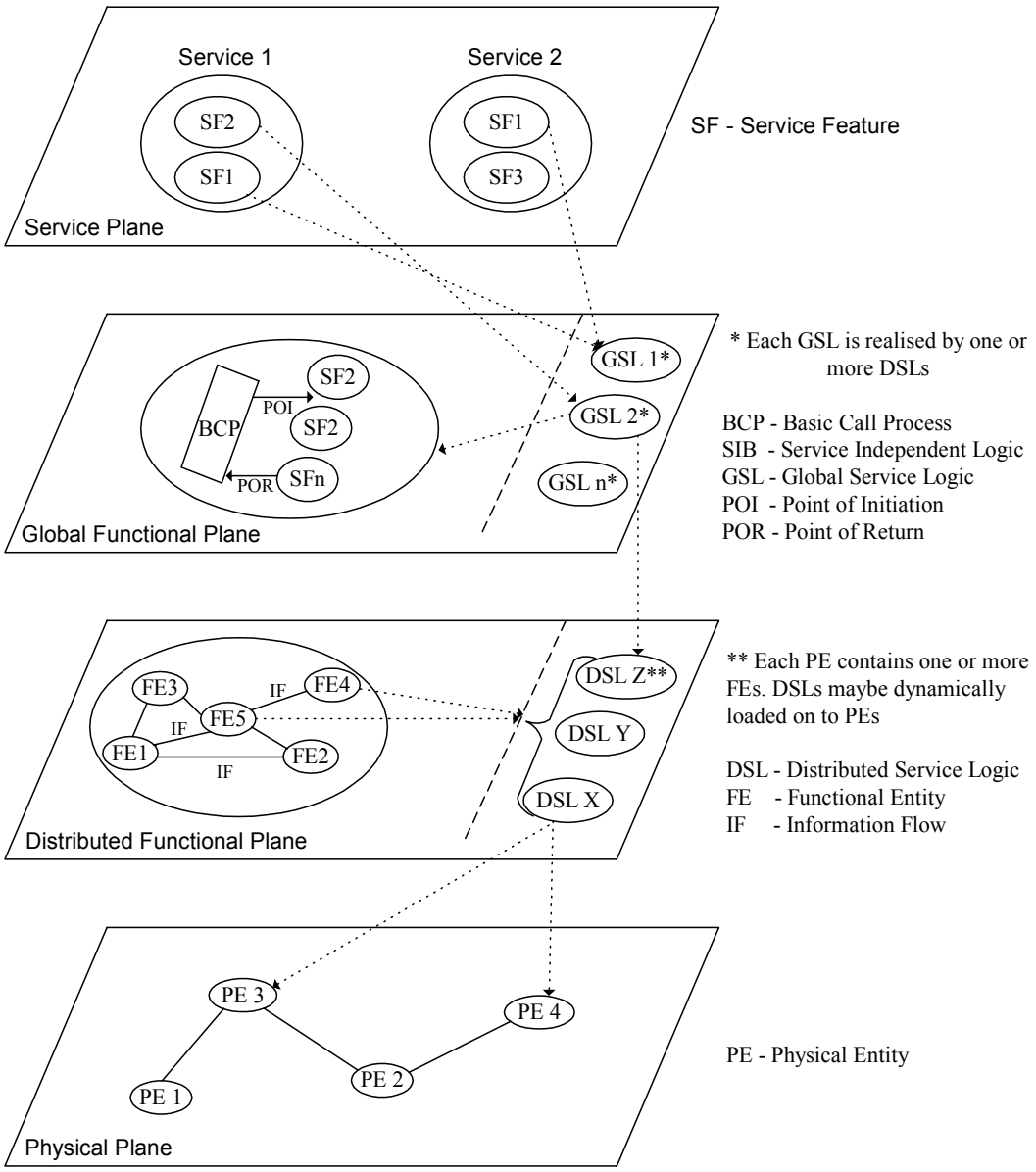


Figure 212: IN Model 4-Plane Structure

Service plane describes services from a user's perspective, where a service consists of generic blocks or service features that make up part or all of a service (e.g., freephone). Global functional plane deals with service creation and is comprised of the SIBs that will be used to create service features. Global service logic defines how SIBs are linked together to form features and how these SIBs interact with another basic SIB known as

the basic call process (BCP). The BCP is the process that optimally supports services that do not require special features and is basic to the processing of all services.

Distributed functional plane defines a set of functional entities that perform specific actions. SIBs are implemented through a specific sequence of functional-entity actions performed by those functional entities. Each functional entity is mapped to a particular physical entity within the network. The information flows defined in the distributed functional plane are implemented in the physical plane through the appropriate protocol, INAP, Intelligent Network Application Protocol.

13.2 IN Components

IN is indeed a platform architecture intended to support a wide variety of services by using common building blocks. Figure 213 shows the 6 components of a general IN.

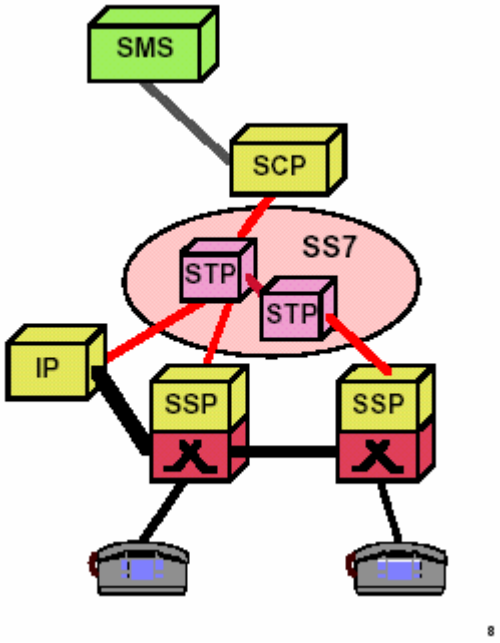


Figure 213: IN Components

13.2.1 Service Switching Point (SSP)

Service Switching Points (SSPs) are stored-program control switches, either local exchanges or access tandem exchanges that are able to interface with the SS7 signaling network. These switches also contain a limited "service (access) logic" required to suspend calls that require special handling. The Service Switching Point recognizes IN service calls and routes the corresponding queries to the Service Control Point via the SS7 network, which consists of Signaling Transfer Points (STPs). Service Control Point commands will be used by the Service Switching Point to further process the call.

13.2.2 Service Control Point (SCP)

The Service Control Point (SCP) is the heart of the intelligent network; it contains the service logic that controls the IN-based services. Its importance can be shown as a single SCP can control IN services of a continental network of public switches using the SS7/IN Application protocol (INAP).

SCP is typically an online, fault-tolerant, transaction-processing database which provides call-handling information as response to Service Switching Point queries.

SCPs handle several hundred transactions per second or more than 100.000 calls an hour, a response time requirement of less than half a second, and an availability of less than three-minutes-per-year downtime for a mated Service Control Point pair.

The SCP is managed with a Service Management System (SMS). SCPs are designed to support multi-service operation.

13.2.3 Service Management System (SMS)

In order to manage IN platforms a Service Management System (SMS) containing the reference service databases is required. Supervision, remote operations and maintenance of SCPs, and (coordinated) software downloading are part of the SMS features. SMS is integrated with a service creation environment (SCE). A new service is developed, assembled and tested using an SCE before being downloaded via the SMS to the various SCPs. Typically SCEs use software components known as service independent building blocks (SIBs), with graphical tools for linking and activating them.

13.2.4 Intelligent Peripheral (IP)

An additional Intelligent Peripheral (IP) may be connected to a Service Switching Point, providing enhanced services/functions, such as announcements, speech synthesizing, speech recognition, etc. under the control of a SSP or SCP. The motivation for the introduction of this network element is an economical aspect, because it might be better for several users to share an IP, when the capabilities of the IP are too expensive to be implemented at all Service Switching Points.

13.2.5 Signaling Network (SS7)

Intelligent networks need to cater for both call-related, out-of-call signaling and require a signaling mechanism logically separated from the bearer channel. The Intelligent Network Application Protocol (INAP), which handles all of INs signaling needs, is a user of SS7 transactions capabilities.

Let us review the basics of signaling. A telephone call requires two basic functions:

1) control signaling function (signaling channel) - setting up and terminate a call,
2) voice and/or data function (bearer channel) - carries information between calling and called parties.

Originally, the telephone network used *in-band* signaling, figure 214, to accomplish both these functions. In-band signaling means that both the control signaling and voice/data functions occupy the same physical circuit or channel. Since voice trunks must be idle during the time call control signaling is taking place, in-band signaling becomes extremely inefficient for advanced calling services.

Figure 214: In-band Signaling

Signaling is just the message-based interaction necessary to initiate certain activities. This signaling network communicates SCPs with SSPs, allowing for communication of information other than normal call setup data. These data communication channels which are separated from the speech channels are called Common Channel Signaling System No. 7, or SS7 for short. SS7 is the hidden signaling network behind telephony activities. With SS7, the signaling control element is no longer part of the voice circuit. Instead, SS7 uses an *out-of-band* signaling channel, figure 215, separating the voice circuit from the routing channel.

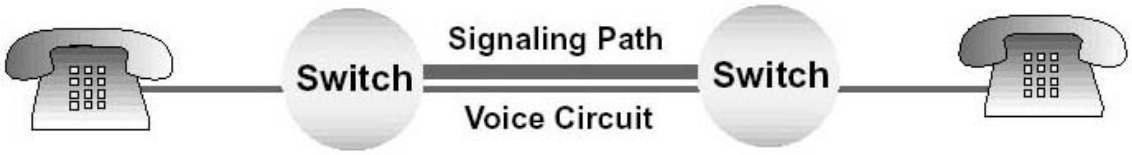


Figure 215: Out-band signaling

SS7 thus provides the underlying network resources so a telecommunication exchange can take place locally or across the world – provided the same signaling system is also being used at the other end of the exchange. SS7 can signal for requested intelligent services, therefore requiring that the network resources respond with the requested intelligent services. Figure 216 shows the structures of SS7 next to that of OSI.

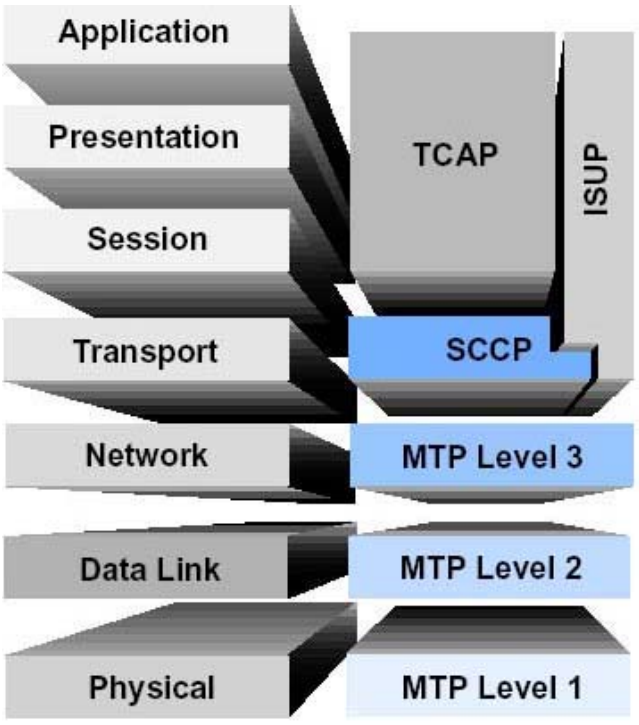


Figure 216: Structure of the 4-layer SS7 contrasted with the 7-layer OSI

Compared with the 7-layer OSI model, the application layer, the 7th layer in OSI and the 4th in SS7, contains the TCAP, Transaction Capabilities Application Part, which has the INAP, IN Application Protocol. The TCAP/INAP is the recommended way of signaling between SSPs and SCPs and, in the future, it will undoubtedly replace all other methods. The Telephone User Part (TUP) is employed by existing networks that use SS7 for normal speech traffic (normal calls). Networks can also use a modified TUP for communication between the SSP and the SCP. Today's business world is marked by the emergence of new, innovative voice services running over converged voice and data networks. These converged networks will also incorporate a wide range of technologies, with the SS7 network acting as the "glue" that holds together many of the pieces.

13.2.6 Signaling Transfer Point (STP)

The Signal Transfer Point (STP) is a vital element in the SS7 network, serving as a signaling hub for the transfer of digital data packets between network nodes. Interconnected to other SS7 elements (signal switching points and service control points) through redundant signaling links, the STP routes messages throughout the network using call information and network addressing structured within SS7 data packets. As the hub in a signaling network, an STP provides centralized network management, facilitating the delivery of intelligent services throughout the network and status concerning the availability of SS7 nodes, routes, and services. Its centralized network configuration allows the STP to serve as a network gateway, screening incoming messages based on message content.

13.3 Intelligent Network 1 (IN/1)

The introduction of IN/1 marked the first time that service logic was external to switching systems and located in databases called service control points (SCPs), figure 217.

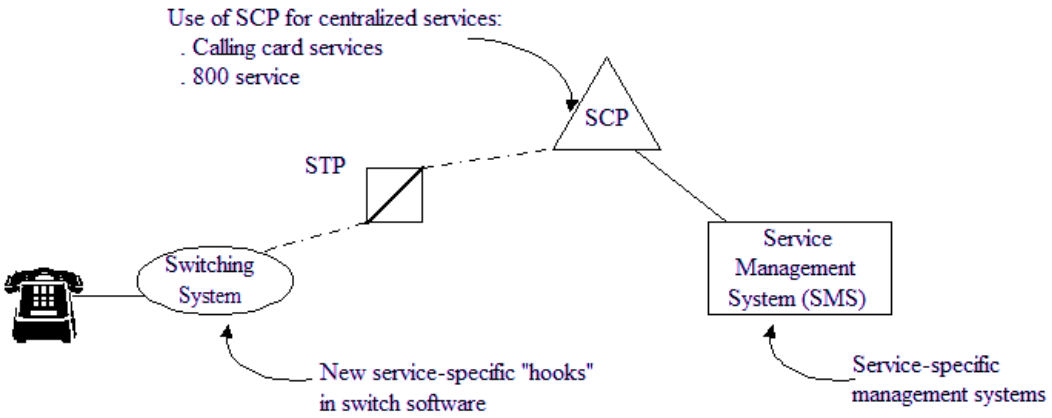


Figure 217: Intelligent Network 1 (IN/1)

IN/1 was a straightforward concept for introducing number-translating services like freephone (800-service) and alternate billing service (ABS), that is, services for which the SCP was a simple “translator” from one telephone number to another. Concepts similar to IN/1 have been implemented in most countries today and represent a natural first step forward implementing intelligent networks. With the IN/1 concept, when a new service was introduced, both the SSP and the SCP had to be updated because IN/1 is a service-dependent technology. For example, an 800 service has an 800-type trigger at the switching system, an 800-service database at the SCP, and an 800-service management system to support the 800 SCP. In IN/1, the 800-service set of capabilities cannot be used for other services (e.g., 900 service). Although the service logic is external to the switching system, it is still service specific.

13.4 Intelligent Network 2 (IN/2)

The IN/2 architecture provides flexibility in the placement of service logic, requiring the use of advanced techniques to manage the distribution of both network data and service logic across multiple IN/2 modules. The idea behind of IN/2 is that when one service is introduced only the SCP needs to be updated. The idea was to simply download the software for the service from the SCP to the SSP via the network.

IN/2 was well ahead of its time, and it was not well received by the equipment vendors because it was too wide-ranging. Some of its advanced features only appeared several years later in ITU-T CS-2 specifications, but even today these remain largely unimplemented.

The physical components of IN/2 are basically the same of IN, that is, SSP, SCP, IP, and SMS. The logical components (software) are the Service Logic Interpreter (SLI), Network Information Database (NIDB), Service Switching (SS), and the Service Creation Environment (SCE)

Very soon, however, it became evident that the IN/2 concept was too difficult to implement in one step, so an interim concept was discussed (IN/1+) [9]. This was also later abandoned and instead focus became fixed on a new concept, Advanced IN (AIN).

13.5 Advanced Intelligent Network (AIN)

The main idea of AIN, compared to its predecessors, is to be service-, switch-, and equipment-independent. Developed by Bell Communications Research in 1991, AIN is recognized as an industry standard in North America. Its initial version, AIN Release 1, is considered a model toward which services will evolve. Meanwhile, evolutionary subsets of AIN Release 1 have been developed. Elsewhere, the International Telecommunications Union (ITU-T), endorsing the concepts of AIN, developed an equivalent version of AIN called Capability Set 1 (CS-1). It comes in evolutionary subsets called the Core INAP capabilities. The progress of AIN has produced CS-2, CS-3, and the last one CS-4. Figure 218 shows AIN components.

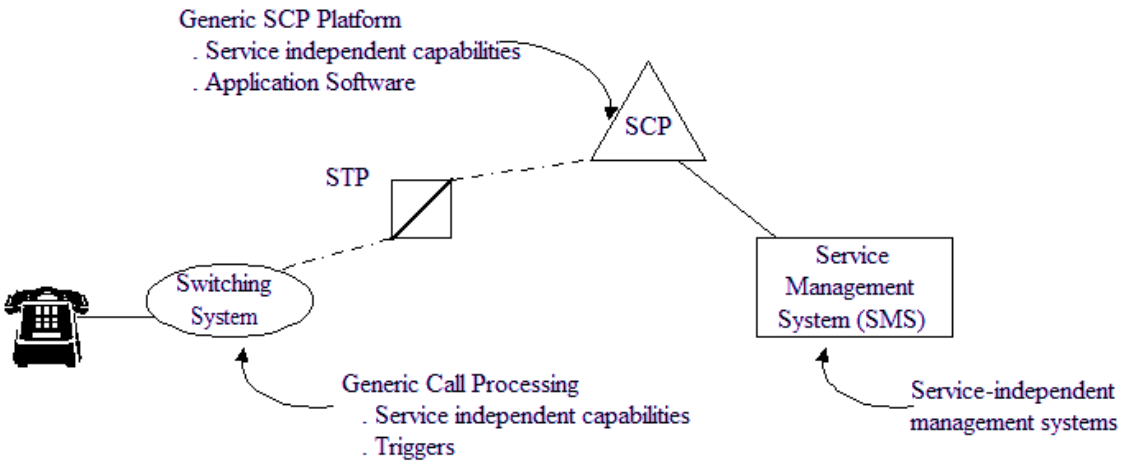


Figure 218: Advanced Intelligent Network (AIN) Components

AIN service-independent software has a three-digit trigger capability that can be used to provide a range of three-digit services (800, 900, XXX, etc.) as opposed to 800 service-specific logic.

Briefly, here's how AIN Release 1 works:

- A telephone caller dials a number that is received by a switch at the central office.
- The switch, SSP, forwards the call over a Signaling System 7 (SS7) network to a SCP where the service logic is located.
- The SCP identifies the service requested from part of the number that was dialed and returns information about how to handle the call to the SSP. Examples of services that the SCP might provide include area number calling service, disaster recovery service, do not disturb service, and 5-digit extension dialing service. In some cases, the call can be handled more quickly by an Intelligent Peripheral (IP) that is attached to the Service Switching Point over a high-speed connection. For example, a customized voice announcement can be delivered in response to the dialed number or a voice call can be analyzed and recognized.
- In addition, an "adjunct" facility can be added directly to the Service Switching Point for high-speed connection to additional, undefined services.

One of the services that AIN makes possible is Local Number Portability (LNP), which is the ability of a telephone customer in the U.S. to retain their local phone number if they switch to another local telephone service provider.

An essential component of AIN is the service creation element (SCE) software. Today, telco personnel at end offices handle service configuration and changes. Over time, the SCE will reside at user’s organization, allowing customers to tailor their services on an as-needed basis without telephone company assistance.

13.6 Convergence IN and IP Networks

We first look back on the development process of networking and communications.

We went from the first generation, during the 70's, which consisted of voice communication, broadcast entertainment, and centralized data networking, past the second generation, the 80's, the Internet, from its infancy as the experimental ARPANET to the ubiquitous Internet, and finally, in the 90's and thereafter, to the third generation of integrated services: the merging of data voice, and entertainment video on a single network infrastructure.

Some expert in the telecommunication field is labeling the first decade of the new millennium as the fourth network generation. It is promoted as being characterized by orders of magnitude increase in network scale, and by the ubiquity of mobile wireless computing devices.

We focus our attention on the third generation: convergence and the Web.

With the advent of IP-telephony, VoIP, the public switched telephone network, PSTN, started to become a subnet of the Internet. Network service providers are scrambling to keep capacity ahead of demand in over-provisioned networks, since the quality of service (QoS) mechanisms to support real-time and interactive applications are frequently being requested.

Figure 219 shows an implementation of a common voice and data network to the desktop.

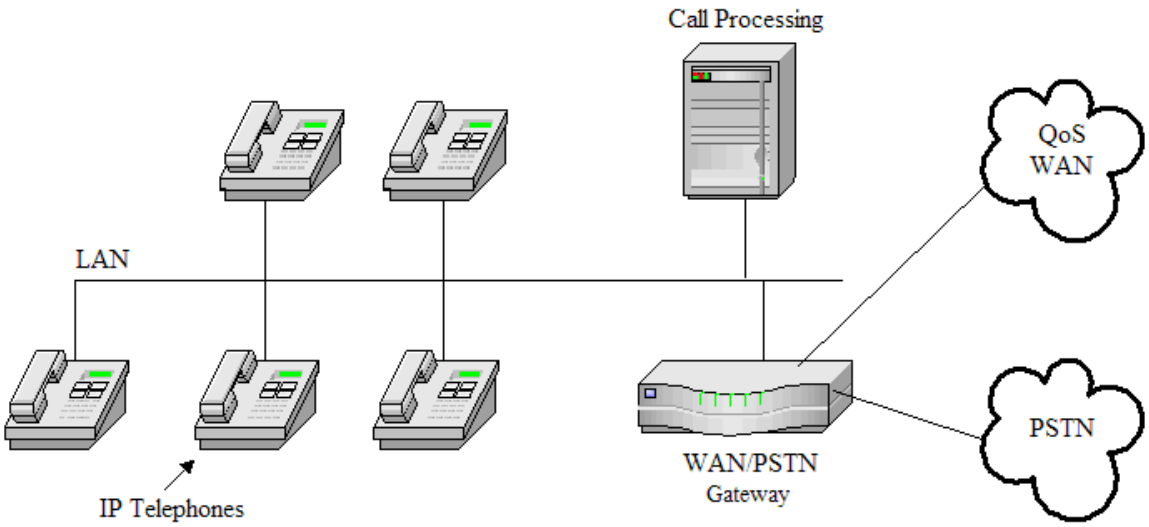


Figure 219: Voice/Data Integration to the Desktop

The third generation is characterized by an IP-based global information infrastructure based on fast packet-switching technology interconnected by fiber optic cable, with IP as the single network layer unifying previously disjoint networks.

The second important characteristic is in the scope of access, with consumers universally accessing the Internet with personal computers via Internet Service Providers (ISPs).

The last relevant characteristic of the third generation is the World Wide Web, which provides a common protocol infrastructure (HTTP), display language (HTML), and interface (Web browsers) to enable users to easily provide and access content.

With all of these benefits, the most popular object used to access the Web is the Intel x86 architecture PC running Microsoft Windows.

Researchers require an open platform with freely available source code for the operating system and protocol stack, and thus UNIX variants are still the platform of choice for the vast majority of networking research.

The lack of open interfaces and source code on commercial routers and switches is a hindrance to the research, development, and trial deployment of new protocols and algorithms.

Future-generation networking will be characterized by the need to adapt to the demands of agile networking, which include rapid response to changing customer requirements, automated design and engineering, lower-cost services, transparent distributed networking, resource allocation on demand, real-time planning and scheduling, increased quality, reduced tolerance for error, and in-process measurement and feedback.

The IN enables a variety of voice features and provides Internet-based connections to control and enhance voice, data, and video, and IP allows new services that are not possible with a circuit-switched networks. The IN infrastructure facilitates delivering those services to existing customers and provides a foundation for development of many of the protocols in the packet world’s products.

14. Introduction to the Intelligent Educational System (IES)

This research proposes an experimental model for new forms of knowledge processing (science learning, teaching, industrial implementation of scientific concepts, marketing and home improvement) supported by intelligent broadband technologies, and hope that such a model can be applied for all kinds of distributed multimedia science learning.

We are going far beyond known uses of Internet by arguing for a system that acts as a 'negotiator' between suppliers of information and consumers that want/need such information. This intelligent system will match the areas where consumers are in disadvantage of with the areas where the suppliers are plenty of. Figure 220 is a conceptual diagram of IENA system, i.e., Intelligent Educational Network Architecture.

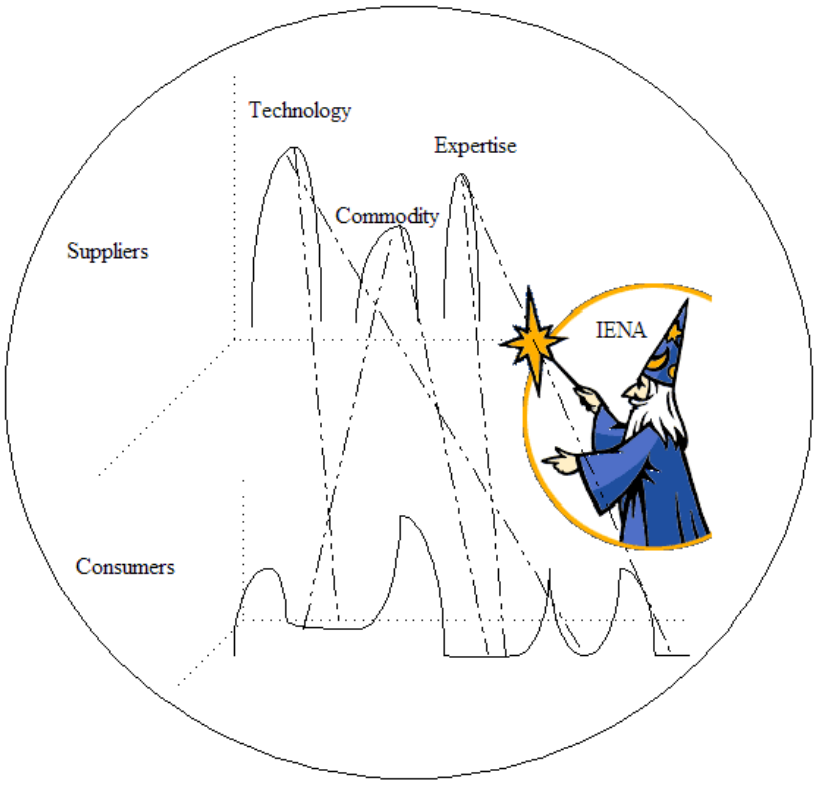


Figure 220: Conceptual diagram of IENA

That is, the peaks and valleys of demand and supply of resources that make us grow in any aspect will be complementing each other for the sake of developing a better humankind.

The system will understand the user's criteria for service request and will direct the user to where the request is qualitatively and quantitatively better serviced. Where there is less, the system will point to the place with the most to be the provider – the system will ensure it. In this respect, IENA helps to make social changes, intelligent decisions regarding the flow of information, and availability of commodities, technologies, and expertise. It will normalize the plain field for all people of the world by giving the ones with less the resources that they need supplied by those with surplus of the same resources. The knowledge bases that hold these resources around the world instruct the system to transfer the knowledge from any supplier to the consumer that is requesting it. Also, information about the consumers will be transferred to the suppliers.

IENA will absorb information from individual, communities, social organizations, government, and cross-country organizations, and based upon criteria for the supply and demand will allocate those resources to the ones that access the system.

An initial step towards a wide reaching system as IENA proposes to be is our IES model that uses the Bellcore's IN enabling technologies. IES has an underlying IP technology with a core engine of IN technology. The IN building blocks allows us to create software applications for student and teacher that provides audio-video and shared data connections, both within a local carrier exchange and long distance public switched networks.

We see Internet as too limiting because it is not a public utility. Our vision should lead to the use of an existing communication infrastructure, the public switched network, for highly interactive multimedia conversations among groups of collaborating students and teachers [10].

The open question that this research is proposing is not to wander in discussions about whether high-bandwidth (>1.5Mb/sec) and multi-functional digital services can be provided, but how such multimedia services will be structured with a convergence of packet- and circuit-switched networks with an intelligent network component embedded into the system. The ultimate goal, once our network model is optimized, is to learn what teachers and learners find useful, in ways that could significantly influence the broadband services and technologies provided by the network that will serve education in the future.

We foresee a future of the provision of research-guided, and educationally sensible uses of the public switched network for highly interactive multimedia communications for science learning/teaching, and individual/collaborative inquiries. If appropriate broadband services can be defined for science education, the scale up will come through first, the commercial availability of such services, and second, their use by public educational institutions.

In summary, intelligent communication networks help knowledge seekers to not only satisfy their present and past, but as well as future demands for their own personal cultural education. In doing so, the collective satisfaction is also achieved. These electronic tutors can complement the traditional face-to-face learning, thus providing a better means to assure knowledge gain at any time and at any place in the world.

14.1 Characteristics of Intelligent Educational Systems (IES)

The main characteristics of IES are:

. As with any existing expert system, IES is not (yet) an “intelligent” system due to the fact that it can not learn from its own experience, a self-learning machine, but its knowledge must be extracted from humans and encoded in the system.

. IES uses domain specific knowledge to provide “expert quality” service in the corresponding problem domain.

. IES’s knowledge gathering and maintenance are the direct result of the interaction between users and the experts that maintain the system.

IES model relies on a hierarchical structure, that is, it is a layered structure. Borrowing from the pioneer OSI model, the function of each layer is as follows:

. Physical: electrical/physical conditions of the network

. Data Link: rules for communication between neighboring nodes

. Network: rules for communication between source node and destination node

. Transport: rules for communication between source and destination processes

. Session: how to establish communication link between source/destination processes

. Presentation: code conversion

. Application: application functions provided to the user

However, unlike the OSI model, IES defines two extra layers: an intelligent processing layer, and the user program layer as shown in the figure 221:

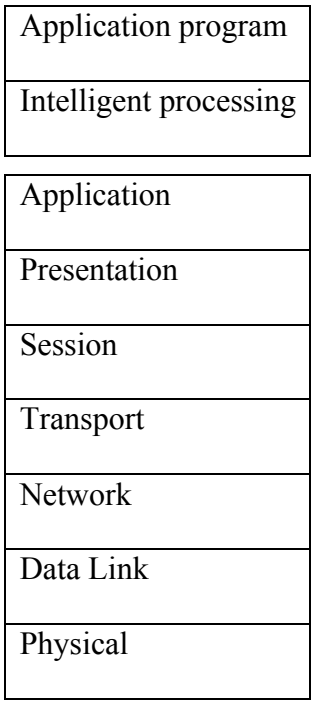


Figure 221: IES layer structure

The locations of the intelligence in the IES are: at the user's terminal application layer and at the switching point in the backbone network. As with the intelligent network, *triggering* is fundamental to the IES, enabling it to recognize pre-assigned knowledge request, and suspend call processing whilst it obtains further knowledge processing instructions from an external service control point.

Our IES is relegated to users that have past a grade school, thus avoiding the need to include elements of individual attention and emotional care that are so common between teachers and students at this level of education. The interaction with the system is via a dialogue processor – a sophisticated program based on the combination of AI, Fuzzy logic, and cognitive science. Such a dialogue processor needs to satisfy different requirements depending on where, what, and who is using it, that is, it has to account for

different point of view of subject matters, student body, and culture. This dialogue processor, the interpreter, must therefore ‘know the user (his/her history) in order to provide the appropriate service.

14.2 Functions of the Intelligent Processing Layer

The functions of the intelligent processing layer are performed by the following systems:

- Knowledge base system
- Natural language processing system
- Media conversion system

The knowledge base system provides problem-solving facilities by using knowledge in a knowledge base. The natural language processing system provides interpreting facilities and language understanding. The media conversion system provides speech-to-character conversion, sentence-to-image conversion, and vice-versa.

14.3 Structure of Knowledge Base System (KBS)

IES, a network-based knowledge processing system, uses domain knowledge stored in knowledge bases. In this research, domain knowledge refers to information classified according to the Dewey Decimal Code system. We will be implementing IES with a simulator, OPNET, and the knowledge bases will be assigned distinct domains based on the Dewey system. As shown in figure 222, the knowledge-base system comprises of a knowledge base, an inference engine, an inference process explanation module, a knowledge-base editor, and a human-machine interface module:

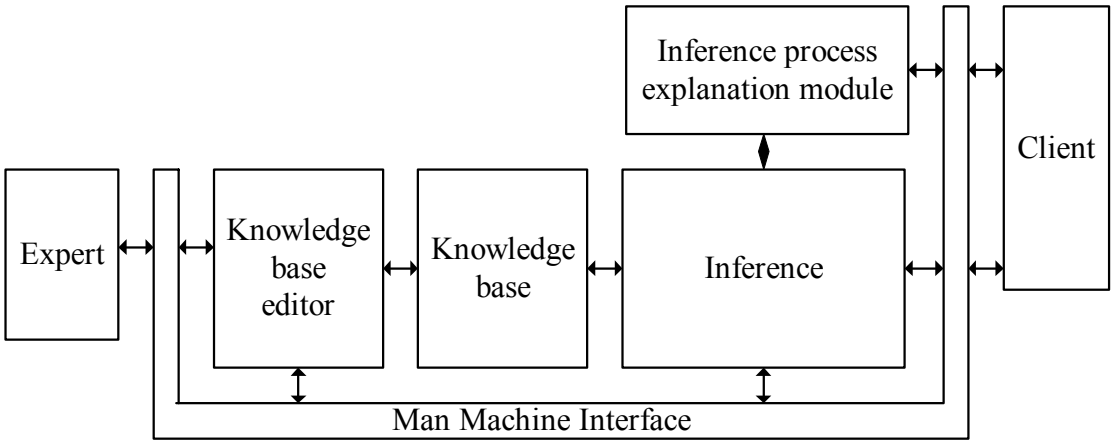


Figure 222: Components of a Knowledge Base System (KBS)

. The knowledge base stores domain knowledge for problem solving.

. The inference engine solves a problem by using domain knowledge.

. The inference explanation module explains the problem solving process.

. The knowledge-base editor provides the facilities for inputting information and editing or maintaining the knowledge base of a specific subject.

. The human-machine interface module, the dialogue processor, interprets users' requests to the system, given in written language, images, spoken language, or gestures. The module interprets the requests and translates them into machine-understandable form.

The heart of IES is the knowledge base, which contains both general knowledge as well as case-specific information. However, the dialogue processor, provides the language for the machine to communicate with the user and vice-versa.

14.4 Knowledge Domain According to Dewey Decimal Code (DDC)

Document classification is the process of assigning a number to an item so as to be able to shelve the item with other items on the same subject. In the United States there are two commonly used classification schemes: the Dewey Decimal Classification and the Library of Congress Classification. Both are used widely and actively updated.

The Library of Congress Classification (LCC) was initially designed at the beginning of the 20th century for the collection of the Library of Congress (LC). The *Dewey Decimal Classification* (DDC) system was conceived to accommodate the expansion and evolution of the body of human knowledge.

Table 25 lists the DDC number ranges corresponding to specific topic domain.

Range	Subject
0 – 99	Computer Science, Information, & General Works
100 – 199	Philosophy & Psychology
200 – 299	Religion
300 – 399	Social Sciences
400 – 499	Languages
500 – 599	Natural Sciences & Mathematics
600 – 699	Technology & Applied Sciences
700 – 799	Arts & Recreation
800 – 899	Literature
900 – 999	History & Geography

Table 25: Top-level Dewey System Classification

We'll adopt the Dewey system even though some studies [41] use the highly detailed international coding scheme UNESCO to avoid the possibility of national or linguistic bias that may occur with the standard Library of Congress and Dewey Decimal schemes.

In our simulation, the traffic flow corresponding to database queries will be classified using the values given by the DDC table. Thus, for example: traffic generated by queries interpreted as being related to Literature is classified as "db-literature", queries whose contents are Languages related is classified as "db-languages", etc.

It will be up to the KCP (Knowledge Control Point) to distinguish distinct traffic in order to route each flow to the corresponding knowledge base(s) (KBs). KCP converts the 'knowledge address identifier' (db-x, x = computer science, philosophy, religion, etc) to the actual physical or logical address of locations in the knowledge bases where that knowledge is stored.

It is worth noticing the similarity between KCP in IES and SCP in IN. The function of the service control point in intelligent networks is to provide access to the IN database and it is connected to the service management system (SMS) which is an user interface to the database. SMS is also responsible for the maintenance of the database.
SCP functions as a translator for the call and instructs the switch how to proceed with the call requesting a service which can be an area number calling service, disaster recovery service, do not disturb service, and 5-digit extension dialing service.

14.5 IES Model Design

In general, network designers always have to face with three aspects of the design:

. Underlying technology might become outdated in a near future

. Growth in network traffic is difficult to estimate

. Network personnel expertise may be missing

One has to keep in mind that networks are neither trivial or cheap; it follows the Truman Reynold's triangle shown in figure 223.

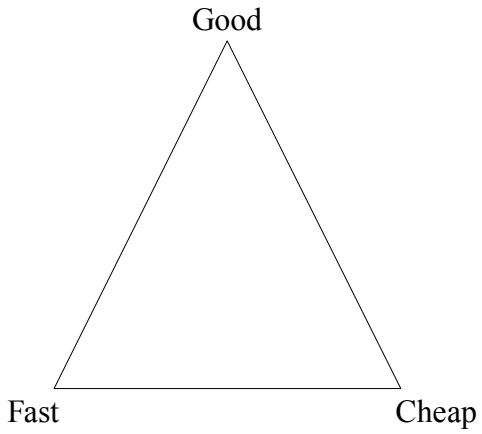


Figure 223: Reynold's Design Triangle

Pick any two vertices of the triangle and then apply the word "not" to the remaining description (vertex). For example, if you want a network that is fast and good, it will not be cheap.

This study uses a simple approach to the IES network design that is called building block approach. The key concept in the building block process is that networks that use a few standard components throughout the network are cheaper in the long run than networks that use a variety of different components on different parts of the network.

14.6 Proposed IES Model

An IES model has to account for the capability of IES to maintain the dialogue between itself and the user as one of the most important functions of the system. A powerful interpreter capable of understanding what the user wants from the system combined with a powerful management system capable of updating IES and making it a self-learning machine are the ultimate goal of building such a system. The IES' interpreter is left out of this paper. It is worth mentioning the idea circulated in some papers [38] of a reference model that describes the way the database queries are processed by an IES with an established knowledge signaling interconnection (KSI) as shown in the figure 224.

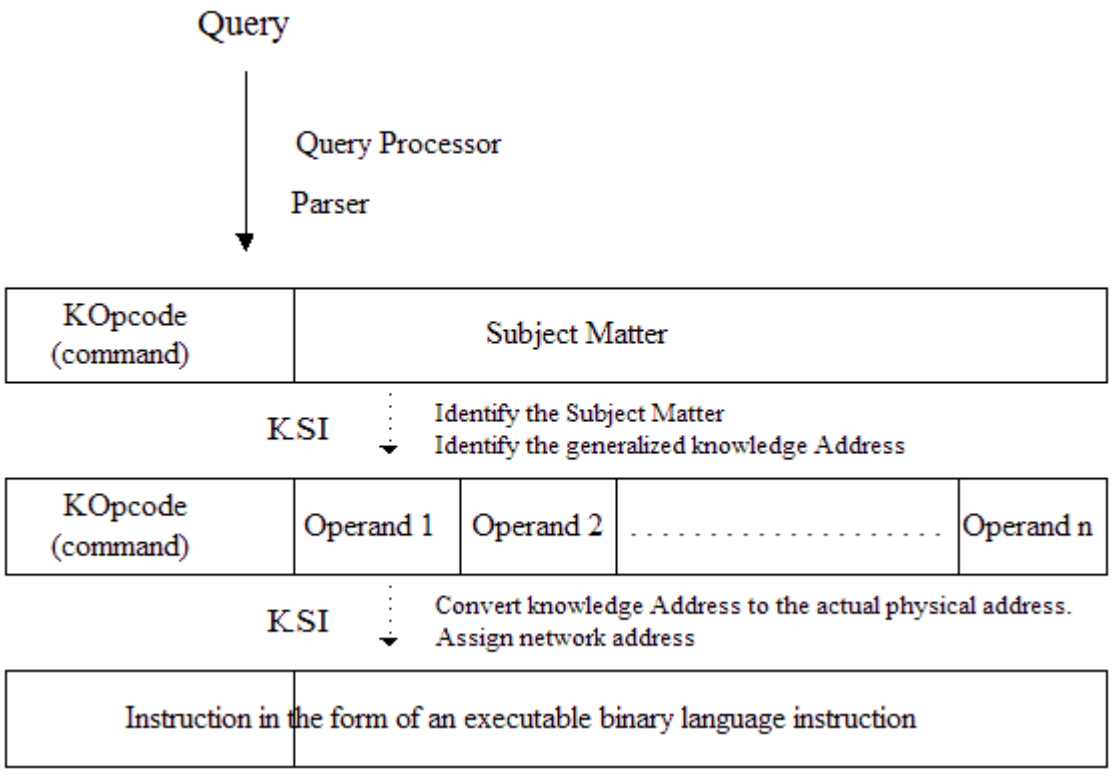


Figure 224: DB Query Processing by an IES

KSI in this conceptual model may define three major entities: Knowledge Query Point (KQP), Knowledge Transfer Point (KTP), and interconnection links.

In our model, the interpreter, which is composed of the knowledge query point (KQP) and of the knowledge transfer point (KTP), passes the instruction (operation code plus an operand) to the knowledge central point (KCP) which then converts the "knowledge address identifier", the domain, to the actual physical or logical address of the location in the knowledge bases where that knowledge resides. Figure 225 shows a diagram of the IES model, implemented in interconnected campus networks.

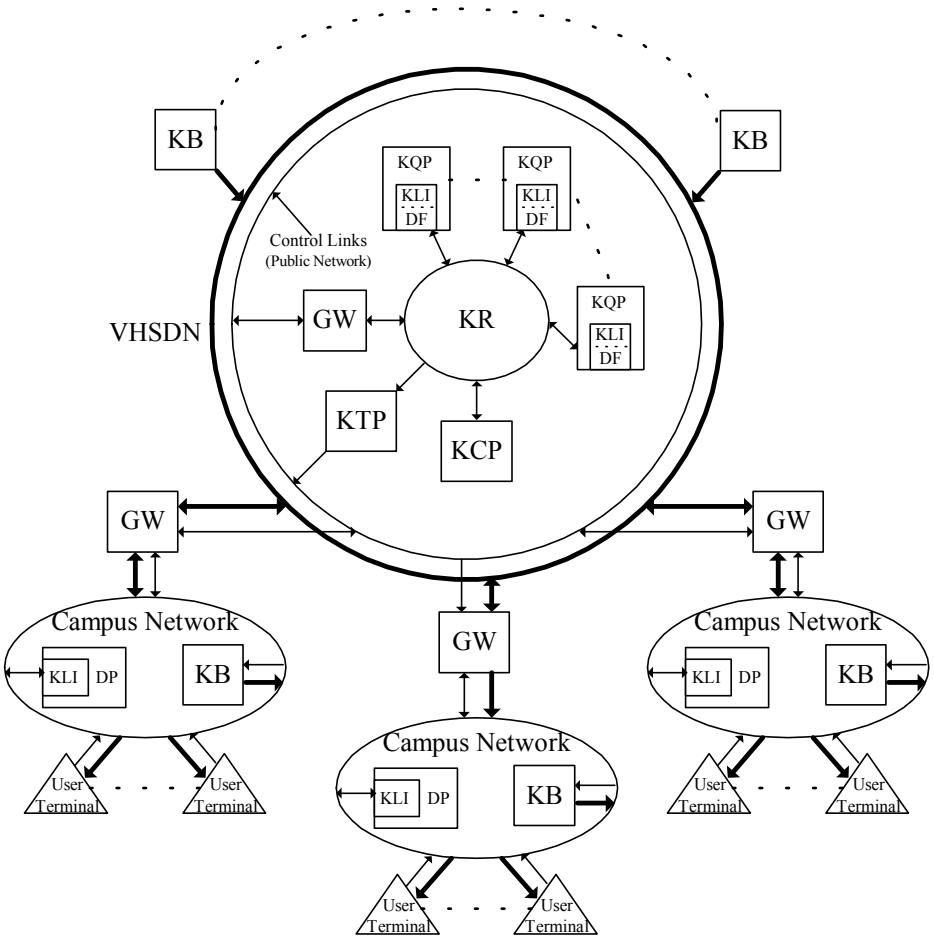


Figure 225: Conceptual Model of an IES

Gateways, GW, in campus networks in cooperation with a remote gateway, monitor and direct the flow of control communications out of campus networks to nodes of a knowledge ring, KR, which is another network, via control communication links, for example, ISDN D channels, provided by a public network. The outflow of queries from user terminals that cannot be handled within the local campus network flows up to KR, and responses to specifically compiled information queries is received in the campus networks.

In each campus network, the authorization and the validity of a request for service (operation code - query/command, and operand - subject matter, which is represented by a hierarchical number, e.g., the Dewey decimal number), are both performed by the dialog processor, DP. Within it, a knowledge logic interface, KLI, formulates and dispatches a binary language instruction to the network components for a response. The information sought is extracted from a knowledge database, KB, to the user's terminal.

A very high-speed data network, VHDN, for the transport of large blocks of data is connected to all campus networks. The control communication links (public network) sends a packet to a specific knowledge base, KB. This is analogous to the signaling SS7 packets of intelligent networks that are exchanged between the switch nodes and the control points. At this point, the knowledge databases transmit large blocks of customized data in response to the opcode via VHDN.

A distributed database differs from our model for not having a signaling network and a control point. It should be noted that the normal flow of information in the IES (dialogue facilitator) is unidirectional while the flow of information is bi-directional in conventional intelligent networks.

When the local knowledge database cannot provide the user with a response, then a remote knowledge base may contain the information requested. These remote knowledge bases are connected to the knowledge ring, KR, which can be any suitable communication medium, such as FDDI, Bellcore's SMDS, or ATT's StarLan.

The KR network contains interfaces for processing the requests, the knowledge query points, KQP, which includes the dialog formatter, DF, and the knowledge logic interpreter, KLI.

The KQP accepts/validates structured queries and infer which knowledge base, KB, to address in order to generate a response. As with the intelligent network, a triggering condition is required for the network to continue the process of responding to the query. This triggering mechanism, (i.e., in intelligent network, the dialing of an 800, 911, or any special service number) for the IES network corresponds to a valid opcode, query or command, and a valid operand, subject matter, which is classified and catalogued according to a subject numbering hierarchy, such as the Dewey's decimal system, the Library of Congress system, the Asian Library system, etc.

Knowledge control point, KCP, is an address translator. It contains a database table with the logic address, the subject matter, and the physical address, the specific knowledge base address that contains the requested information. Once the address translation is done, the packet request is sent via the knowledge transfer point servers, KTP.

To provide fast and intelligent communication services, the VHDN network switches can be upgraded to process call signals and decide which knowledge base the response should come from. This switching module checks on its trigger table for more information to route the call through the signaling network to the specific knowledge base server containing the information about the type of the call or application.

In this model, knowledge messages (commands) can be routed through the networks to perform knowledge management (setup, maintenance, termination) as well as network service functions. The KSMS ensures up-to-date data for IN translations are made available wherever it is needed by the service control logic. Up-to-date versions of frequently used data may need to be held locally at all KSCP sites, although it may not be efficient to distribute local copies of rarely used data to every control point in a large network just in case it is needed.

From the current situation of computer systems, IES is a utopia, an imagined tutor-robot with the first real sign of human intelligence – the capability of self-learning. Obviously, this research is further from being the groundbreaking work of such system, but it wants to be one of the researches that try to shine some light onto the quest of an actual IES.

The type of such network envisioned by this research may be futile and overpriced for environments where the emotional involvement between the student and teacher is high and education is much more than the textual or graphical exchange of information as it occurs in elementary and secondary schools. For this reason, the proposed model is more appropriate for research institutes and graduate level universities.

14.7 Simulation of IES

We chose again OPNET as our simulator - its modeler operation provides an extensive library of nodes, links, protocols, and traffic objects. Nodes and links are organized so that separate subnets represent the different knowledge bases that hold domain-specific data. The nodes, links, and subnets are similar to those used in our NCN and ICN networks.

The Poisson probability distribution is selected for modeling the call (data) traffic, inter-arrival times, holding times, and data size. Poisson is the preferred statistical distribution for stochastic events – the current occurrence does not have anything to do with the previous occurrence and vice-versa. The simulator generates network traffic coming from applications running in each workstation that are very common in campuses: Web browsing, email, FTP, and database queries

To simulate the processing of the opcodes (query/command) and operands, port numbers in the intelligent switches (KCP) will be assigned according to the application and their domain. Thus, by implementing intelligence at the port level versus system level as most of the existing switches do – where routing decisions are made only on the basis of prevailing network conditions, we can enable virtualization, quality of service, and replication to happen within the network.

The next three steps will conclude our simulation: (i) network topology, (ii) network traffic, and (iii) network and simulation operations. We will then evaluate the results and make inferences.

14.7.1 The Network Topology

The network topology is analogous to our national campus network (NCN). But now we will call it the intelligent national campus network (INCN). INCN, figure 226, consists of a specific subnet that is replicated at 6 nodes geographically located at New York, Miami, Houston, Las Vegas, Seattle, and Chicago.

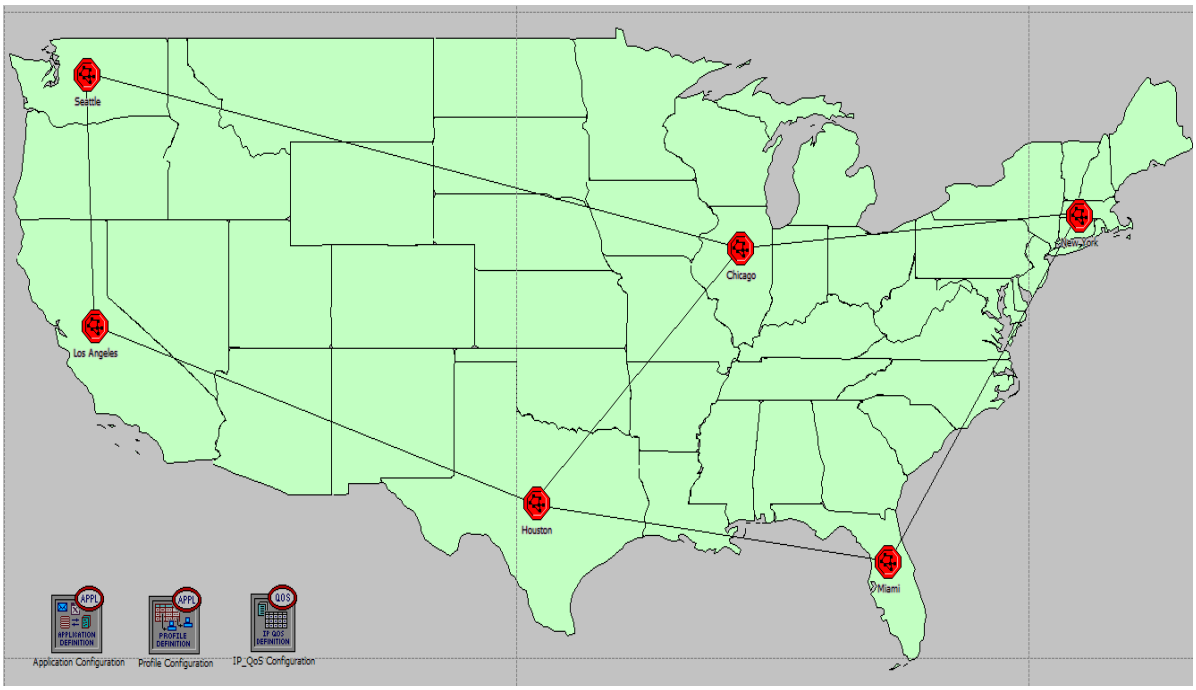


Figure 226: Layout of the Intelligent National Campus Network (INCN)

These nodes are connected via optical fibers with speeds of up to 155 Mbps (OC-3), although, in reality, the data packets flowing through the links are the payloads for SONET packets, and thus the real speed is around 149 Mbps to account for the SONET overhead. These are PPP duplex links (Point-to-Point Protocol) establishing direct link between the nodes running IP.

Each node is composed of individual workstations, switches, and routers as shown in figure 227. The main functionality of a node is to act as: (i) a place where network traffic may originate, (ii) a place where network traffic may terminate, and (iii) a place where network traffic may be switched through. Each node is considered a subnet of the entire network. Figure 227 shows the subnet of a particular node, New York.

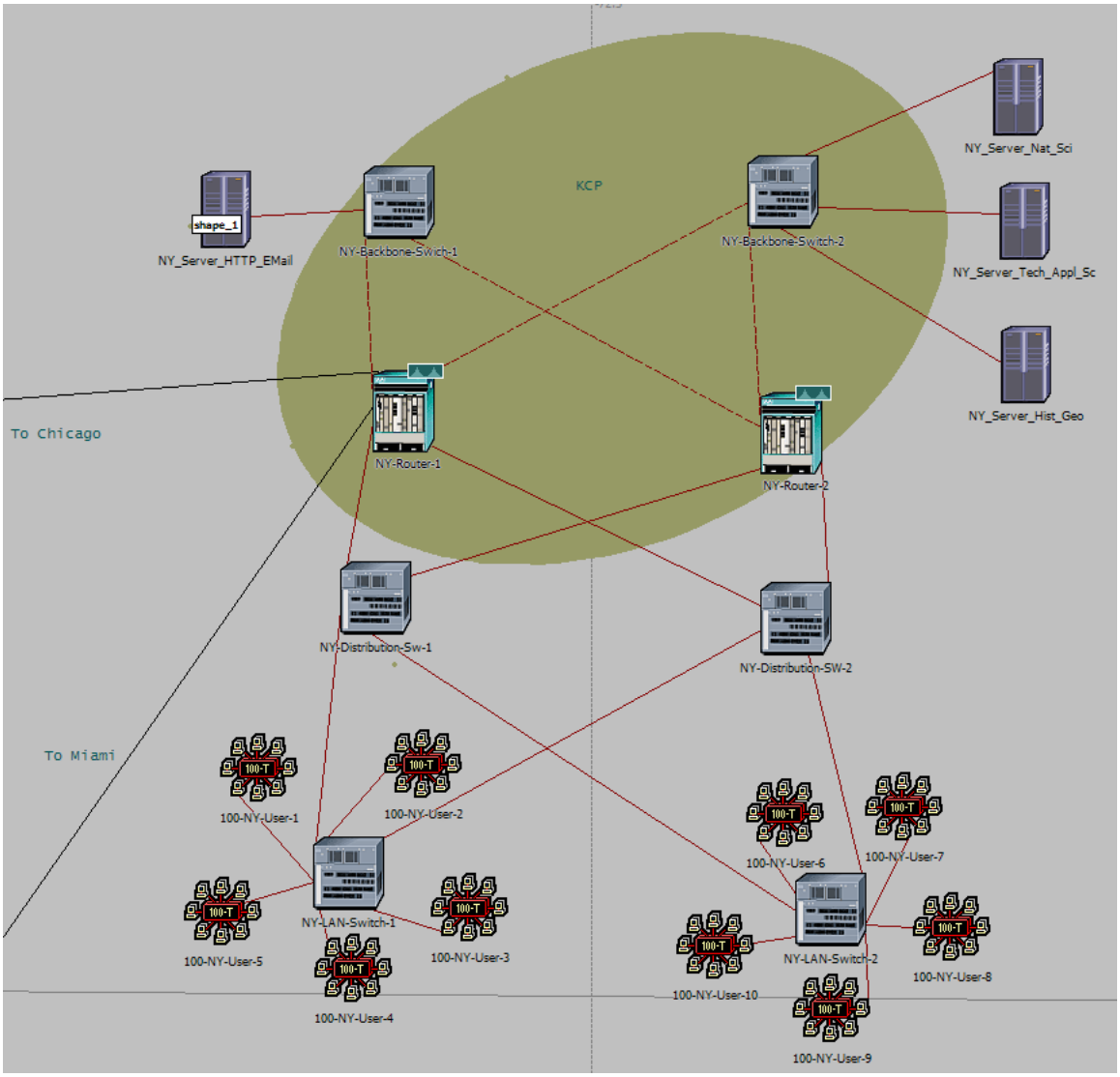


Figure 227: Subnet with interfaces acting as KCP

The subnet is completely faulty-tolerant due to redundancy built-in to each component. Therefore, if a switch and/or a router go down, the network continues to be operational.

The subnet shows a darkened ellipse indicating the KCP, which is composed by intelligent servers, switches and routers. The KCP, which corresponds to the KSCP in our model, have been programmed to be the network agent providing appropriate routing for database queries packets that have been marked and labeled with any of the top domains of the Dewey Decimal Code system.

Each LAN corresponds to 100 workstations with the same hardware characteristics connected to the same LAN-switch. Therefore, INCN correspond to a network providing services to 6,000 users. Each user (workstation) generates the traffic on the network.

The traffic comes from the applications that the workstations are using. The typical applications used in campus networks are: Web browsing, email (sending and receiving email), file transfer, and database queries and updates. The link used to connect the nodes is OC-3, which is the one that resulted in better performance of our networks in the first part of this research.

The links between the devices are:

- OC-3 (155 Mbps) Router-to-Router
- 100BaseT: Workstation-to-Switch
- 1000BaseT: Switch-to-Router, Switch-to-Switch

14.7.2 Representing the Network Traffic

We will use a synthetic traffic, i.e., OPNET's statistical traffic sources, to represent the traffic. An alternative way would be to read in real data from a live network, but this is beyond the scope of this thesis.

OPNET uses a generic network application model to generate typical application traffic patterns. This is the application model, also called the standard network application model. Most of the standard OPNET client-server node models embed the applications model to model the traffic they generate and the way they treat received traffic. Each application can be enabled or disabled on the client nodes through use of one or more profiles, and each can be specified as a supported application service type on the server nodes.

The typical campus network's applications are: Web browsing, E-Mail, ftp, and database. We are interested specifically in traffic generated by database application which enables the user to store and/or retrieve information. Database operations in OPNET are divided into two categories: (1) a database entry, and (2) a database query.

A database entry results in a fixed amount of data being written into the database, while a database query results in the client issuing a query and the server responding with some data. We limit our study of simulation results to only those of database queries since the objective is to verify the functionality of our intelligent network regarding knowledge queries whose answers reside in knowledge bases. As it was done in the first part of the thesis, we assume exponential distribution for interarrival time and Poisson distribution for packet size.

The default transport protocol for all of the applications is TCP.

Table 26 describes some of the important database attributes.

Attribute	Description
Transaction Mix	Queries/(Queries + Updates)
Transaction Interarrival Time (sec)	Time between transactions
Transaction Size (bytes)	Average size of an entry or a response to a query
Symbolic Server Name	Symbolic name of the server the client contacts
Type of Service (ToS)	QoS parameter priority to an application's traffic
RSVP Parameters	RSVP parameters for making bandwidth reservations

Table 26: Database Attributes

Figure 228 show the settings for one of the database applications.

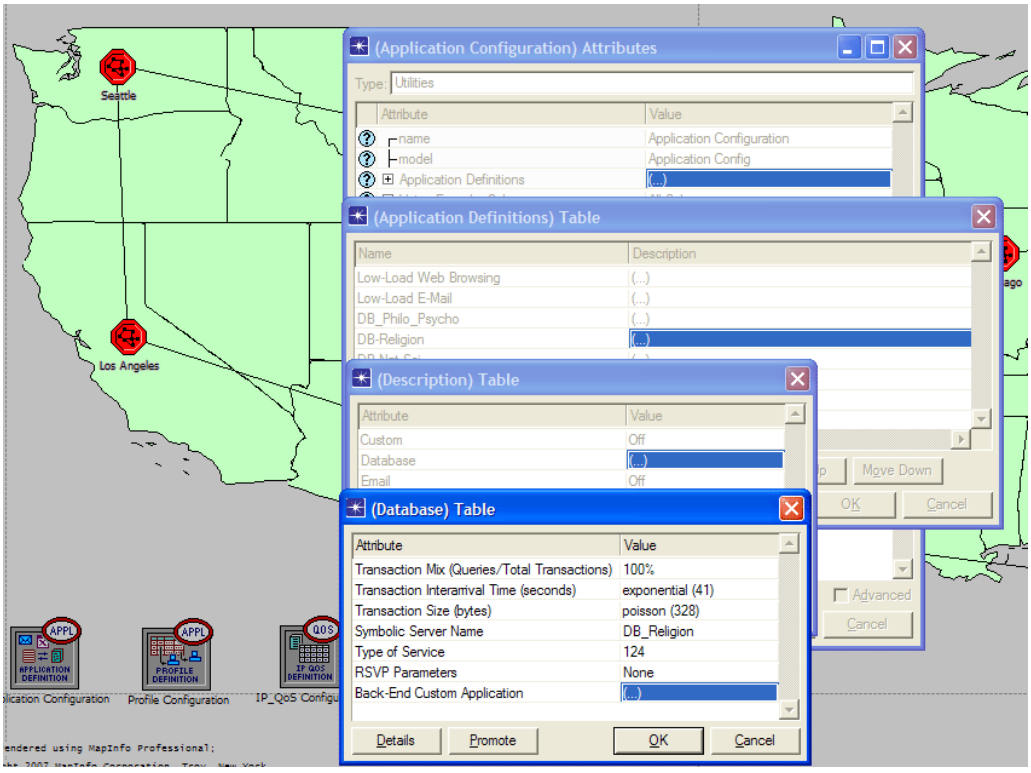


Figure 228: Database Application Settings

There are 10 top domains on the Dewey Decimal Code system which were listed in the table 25. We are using the finding of studies conducted on academic subject searches that have the most online impact [111, 112] classified according to the DDC. These studies conclude that Computer Science, Information, and General Works searches are the top type of searches conducted online with 30% of all DDC-classified searches. And, on the other hand, Religion and Languages searches are only 1% of all of the searches. Table 27 shows the results of these studies.

DDC Range	Subject	Percentage
0 – 99	Computer Science, Information, & General Works	30%
100 – 199	Philosophy & Psychology	2
200 – 299	Religion	1
300 – 399	Social Sciences	20
400 – 499	Languages	1
500 – 599	Natural Sciences & Mathematics	15
600 – 699	Technology & Applied Sciences	25
700 – 799	Arts & Recreation	2
800 – 899	Literature	2
900 – 999	History & Geography	2

Table 27: Online searches by DDC

OPNET suggests setting the minimum value for database queries interarrival time to 12 sec and their maximum transaction size to 32,768 bytes in order for the simulator to be

able to trace capture optimally packets taken simultaneously at the client and server and filtered by application type. With these two values plus the percentages for each subject on table 27, we can then compute the values for interarrival times for each DDC and the corresponding transaction size. The top domain, Computer Science, Information, & General Works, is assigned an interarrival time of 12 seconds and a transaction size of 9,830 bytes. While the bottom domains, Religion and Languages, each has an interarrival time of 41 seconds and a transaction size of 328 bytes. Figure 229 shows the profile configuration table for the applications.

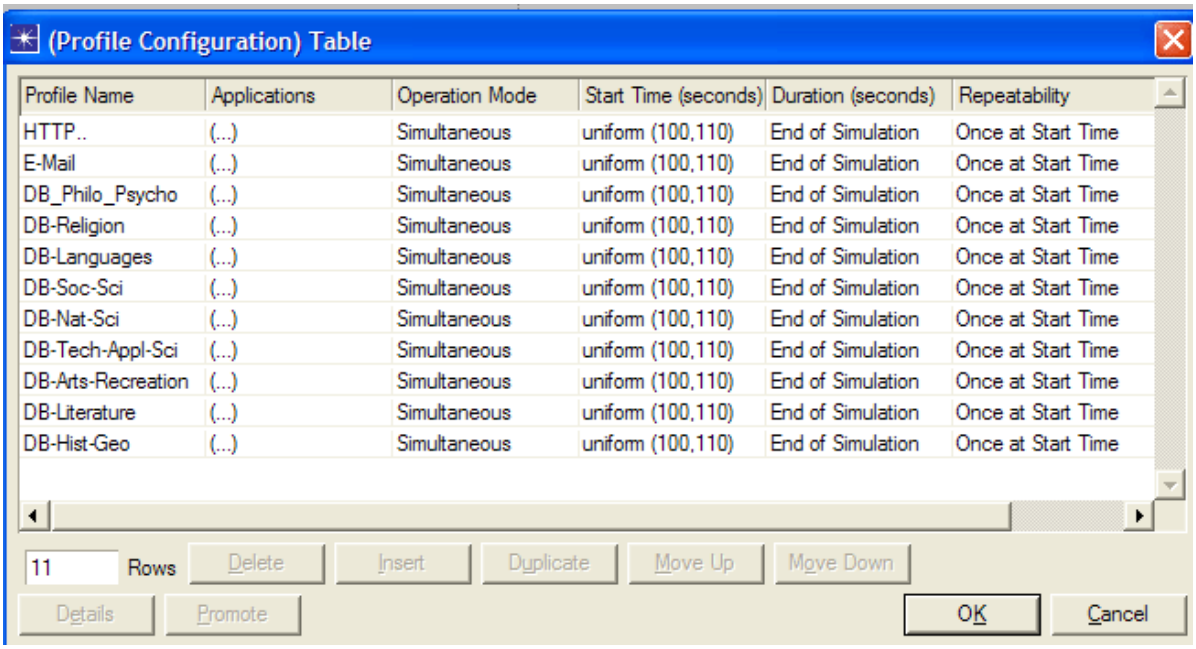


Figure 229: Profile Configuration

In order to complete 10-minute simulation runs with the existing computing resources:

(i) assign low loads for Web browsing and E-Mail traffic,

(ii) disallow FTP traffic,

(iii) knowledge bases in only two nodes (LA & NY) , and

(iv) assign smaller packets for DDC-domains according to their predominance.

The simulations were run in a Toshiba laptop with a Pentium 4 CPU, cycling at 1.7 GHz, and with 512 MB of RAM. The average time it took to run a 10-minute simulation was around 12 hours.

OPNET's default values for low load Web browsing traffic are:

- Page Interarrival Time (sec): Exponential (720)
- Page Properties: 10 pages of 500 bytes each containing 5 small images per page
- Type of Service: Best Effort (0)

OPNET's default values for low load E-Mail are:

- Send/Receive Interarrival Time (sec) Exponential (3600)
- Send/Receive Group Size (bytes) Poisson (3)
- E-Mail Size (bytes) Poisson (500)
- Symbolic Server Name: Email server
- Type of Service: Best Effort (0)
- RSVP Parameters None

Knowledge Bases:

- NY: History & Geography, Natural Sciences & Mathematics, and Technology & Applied Sciences.
- LA: Philosophy & Psychology, and Religion.

The workstations were set up to generate traffic of these 7 applications: Web browsing, E-Mail, DB-Phylo&Psycho, DB-Religion, DB-Hist&Geo, DB-NatSci&Math, and DB-Tech&AppliSci.

The main focus is on the traffic generated by the database queries, which the KCPs (knowledge control points) decide where the traffic flow corresponding to the client's request should be routed to and where the corresponding knowledge base's response should be routed to.

Background Traffic and Daily Network Traffic:

One more aspect of the traffic is the presence of background traffic which simulates real situations having unwanted traffic present in the network causing queuing delays on intermediate devices.

OPNET traffic generator is used to generate a source of background traffic. Actual network analysis [37] shows that traffic rises gradually over the course of the day as employees arrives and begin work. This pattern can be modeled using background link utilization. The daily network traffic analysis shows that from 9am to 10am traffic rises about 10%. Then, from 11am to 12am it goes down about 30%, and from 1pm to 2pm it goes back up 30%. Finally, from 3pm to 4pm, the traffic goes down 10%. Therefore, to simulate the traffic behavior throughout an ordinary day (our 10-minute simulation runs), the background load was incremented every 200 seconds from 10%, then to 20%, and finally to 30% of the total traffic.

QoS:

The traffic is also influenced by quality of service, QoS, of the network. We do not want the uncertainty of the delivery of the packets so characteristic of IP networks. Due to its IP-nature and reliance on IN-features, the intelligent educational system's dialogue has the same delay requirements as VoIP exchanges. QoS issues are directly addressed in IP version 4 (IPv4) basically by defining the "Type of Service" (ToS) byte in its header. ToS values represent a session attribute which allows packets to be processed faster in IP queues. It is an integer between 0 - 252, 252 being the highest priority, see table 28. ToS assignment at the client is not affected by the ToS value specified at the server.

ToS	Value
Best Effort (0)	0
Background (1)	32
Standard (2)	64
Excellent Effort (3)	96
Streaming Multimedia (4)	128
Interactive Multimedia (5)	160
Interactive Voice (6)	192
Reserved (7)	224

Table 28: ToS Values

The database traffic are treated as having "Excellent Effort" to expedite these packets through the network and leaving the other traffic (Web browsing, E-Mail, and FTP) with

Best Effort treatment. Therefore QoS is incorporated into the simulations. Figure 227 shows an example of the settings for a database query application. The settings shown is for DB-Religion and it has 100% transaction mix, Exponential (41) as the interarrival time (when the next database transaction will start), Poisson(328) for transaction size, a server named DB_Religion for contacting, and Type of Service of value 124 corresponding to Excellent Effort in the three areas Delay, Throughput, and Reliability.

Queue and Buffer:

It is advisable to establish a queue size so to prevent that the queue grows indefinitely because the combination of arrival rate and packet size may exceed the service capacity. The maximum queue size is set to 500 packets and database queries are set as having low latency queuing which enables the use of a single, strict priority queue for delay-sensitive traffic. Traffic in this queue gets the highest priority, and only if this queue is empty, are other queues allowed to send traffic according to the adopted queuing mechanism.
This study opted for WFQ (weighted fair queuing) with WRED (weighted random early detection) for congestion control. They have the ability to recognize and act upon congestion on the output direction of an interface by dropping packets randomly. WRED takes into account the type of service (TOS) by increasing the dropping probability of low-priority traffic. The buffer capacity is set to 1,000. While this value is not reached, a queue can store packets, whatever the size of the queue. When this value is reached, the interface is in a state of congestion. This means that a queue which has more packets than its Maximum Queue Size will drop incoming packets according to WRED until either Maximum Number of Packets or Maximum Queue Size drops.

14.7.3 Network Simulation Operations

The network operation basically identifies the routing algorithm, i.e. how the messages travel from the source to the destination. This is necessary if multiple paths between nodes exist. In our project, we do have multiple paths, so we do need to concern ourselves with the details of the different algorithms.

The IP packets are routed across the INCN using RIP routing information protocol. RIP, the most common routing protocol in the Internet, is a distance-vector dynamic routing protocol. RIP uses hop counter as a routing metric. Routers using RIP simply count the number of computers in the possible routes to the destination and select the route with the least number of hops (a hop is one circuit). These routers exchange routing updates and thus build routing tables dynamically.

OPNET also provides for the capability to configure these routers to do content-based routing, where the packets once they are properly labeled can be routed to an appropriate destination by matching content elements against pre-established policy rules.
It should be emphasized that content-based routing is more similar to Cisco's network-based application recognition (NBAR) [38] than to the more complex deep-packet inspection (DPI) technique [40].

In our study, we are requiring that the KCP machine examines packet types (according to the specific application they belong to) and translates this examination into the

appropriate knowledge base destination address. KCP translation is activated by the type of application the traffic flow has.

The proposed architecture for a service logic interface, our KCP, is one where an overall QoS signaling combined with content-based routing is consisted of three layers as depicted in figure 230.

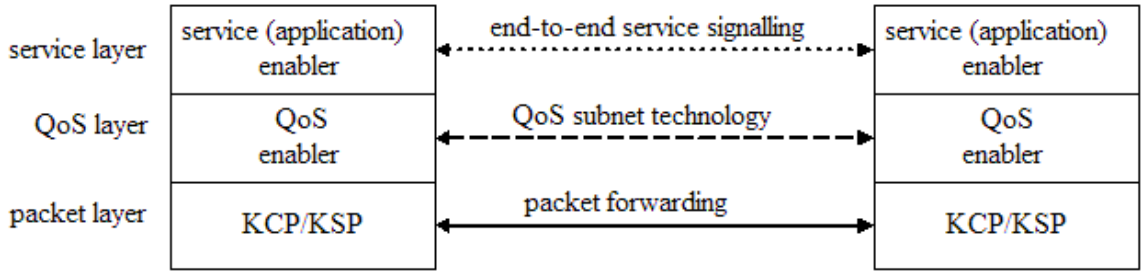


Figure 230: Conceptual View of Packet Forwarding with QoS & Content-Inspection

The signaling protocol for database queries related to knowledge needs to be defined. However, in the case of a query related to intelligent services (IN services), the already existent SS7 can be used.

The algorithm used, shown in Fig. 231, is a series of steps that is necessary to accomplish the task of sending a database query over the network and receive a response from an appropriate knowledge base. The traffic flow of database queries, as opposed to the traffic generated by HTTP and E-Mail, is treated differently (differentiated via DiffServ) by the interfaces as it leaves the user's workstation, arrives at the final destination, and returns with a response to the user's workstation. The KSCP is a combination of knowledge service switching (KSP) and knowledge control point (KCP).

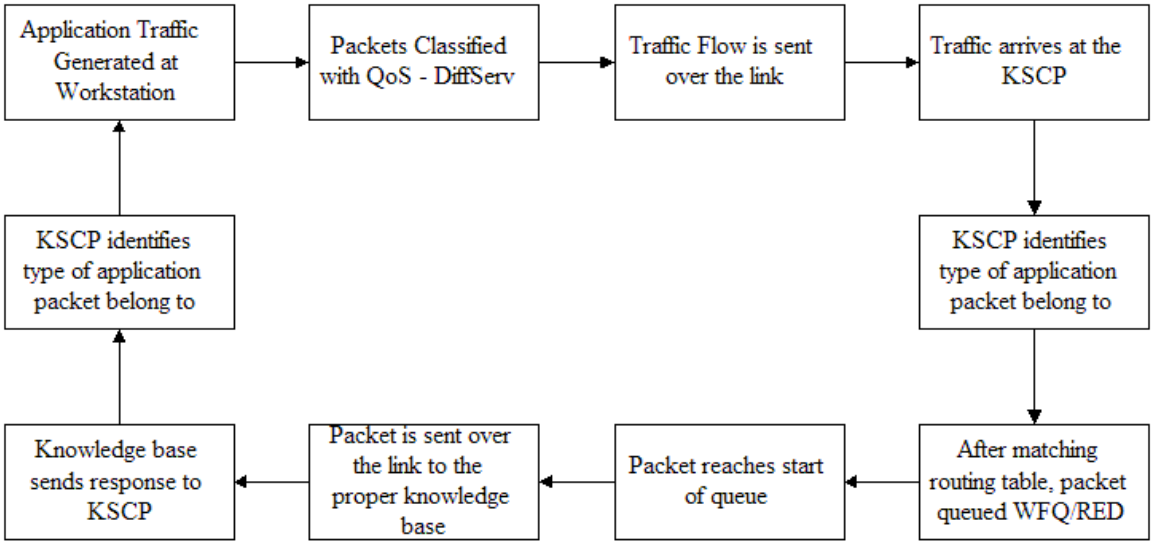


Figure 231: Algorithm for Processing Database Queries

The KSCP machine combines the functions of the service switching point and the functions of the service control point. It is an intelligent machine that reduces the programming effort to do the knowledge processing. These can be the only machines that need to have their logic programs upgraded to accomplish the task of identifying the packets and routing them to the appropriate knowledge bases, either the ones that are local to the user's network or are found in another interconnected intelligent networks.

As for the simulation parameter, it is important to set the simulation run length and the warm-up period. The simulation run length is 10 minutes which enabled us to observe a large sample of data implying a considerable amount of accuracy on the evaluation of the results. The warm-up period is the first 100 seconds. The applications are started after the warm-up period with the probability between a minimum 100 seconds and a maximum 110 seconds of the simulation run.

14.7.4 Running the Simulation

The simulation of an intelligent educational system (IES) generates performance statistics on the network that carries the system which consists of a user interface, inference engines, and knowledge bases with specific subject matter.

The focused point is to understand the behavior of the knowledge control point, KCP, shown in figure 226, more broadly as KSCP, which has an additional function besides controlling packet forwarding: additional processing and deeper examination of incoming packets. The KCPs (one at each node) do not simply forward packets; they act intelligently in deciding when and how to forward them to the appropriate knowledge servers. These knowledge control points are programmed to use the routing algorithm to decide how to forward packets based on their domain, and when to forward them based on the queue management. There are many factors that must be taken into account, including performance requirements and types of database queries (application-specific) factors that must be evaluated when applying the chosen technology.

14.7.5 Scenario INCN

INCN is made of 6 nodes interconnected via OC3 duplex-link:

NY, Miami, Houston, LA, Seattle, and Chicago.

Each of these nodes has the same type of subnet shown in figure 226.

A total of 6,000 users generating traffic from the following 7 applications:

. HTTP (low-load), E-Mail (low-load), DB-Religion, DB-Philo&Psycho, DB-Hist&Geo, DB-NatSci&Math, and DB-Tech&AppliSci.

Again, due to limited computing resources, only NY and LA have DB servers besides HTTP and E-Mail servers, all the other nodes only have HTTP and E-Mail servers.
These are the Knowledge Bases in the two nodes:

- NY: Technology & Applied Sciences, Natural Sciences & Mathematics, and History & Geography.
- LA: Religion, and Philosophy & Psychology.

Also, in each node, the knowledge control points, a combination of switches, routers and servers, function as the traffic controller for the entire subnet. All of the traffic that comes from inside and from outside of the subnet is ultimately directed by the KCP (intelligent device housing the service logic, the switching, and the management operations). KCP has also access to the knowledge bases.

Our traffic is subjected to QoS and the interference of a background traffic.
We have 3 choices of the type of statistics that can be collected during the simulation:

1. Global Statistics
2. Object (Node) Statistics, and
3. Link Statistics

You can collect statistics from individual nodes in your network (Object statistics), and/or from the entire network (Global statistics), and/or from the links that connect the nodes of the network (link statistics). Global statistics allow you to examine network behavior for the entire network model.
Before we evaluate the statistics it is worth viewing the way traffic (blue arrow lines) flowed through the network. Figure 232 shows the flow through the NY node. Notice that

the redundant part of the network is set aside while the main interfaces do the work of carrying the main traffic.

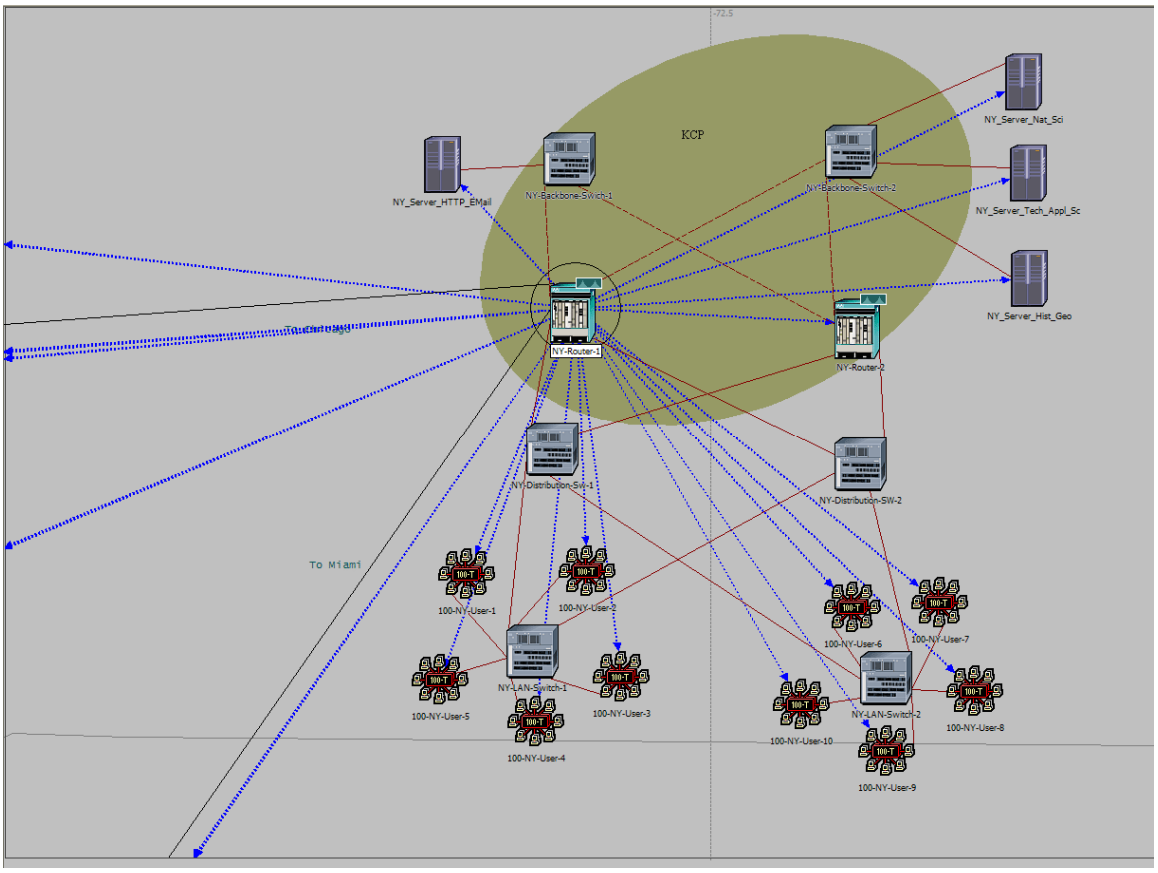


Figure 232: INCN Traffic Flow through New York Node

It is also important to notice the expected routing behavior of the SCP, showing the correct delivery of the packets to their appropriate destination knowledge base through the identification of the packets via their associated type of application.

The following tables and graphs shown below display, for each application, the global, the node, and the link statistics with their descriptions and the corresponding graphs generated by the simulation run.

14.7.6 DB-Query Statistics

Global Statistics – DB-Query	
Metric	**Description**
Response Time (sec)	Time elapsed between sending a request and receiving the response packet. Measured from the time when the Database Query Application sends a request to the server to the time it receives a response packet. Every response packet sent from a server to a Database Query application is included in this statistic.
Traffic Sent (bytes/sec)	Average bytes per second submitted to the transport layers by all Database Query Applications in the network.
Traffic Received (bytes/sec)	Average bytes per second forwarded to all Database Query Applications by the transport layers in the network.

Table 29: Global Statistics for DB-Query

Node Statistics – DB Query	
Metric	**Description**
Load (requests/sec)	Rate at which any DB Query requests arrive at the server. Note that these requests could belong to different sessions maintained at the server. Also note that requests for the same session is queued until the first request is completed.
Traffic Sent (bytes/sec)	Average bytes per second submitted to the transport layer by the Database Query application in this node.
Traffic Received (bytes/sec)	Average bytes/second forwarded to the DB Query Application by the transport layer in this node.

Table 30: Node Statistics – DB Query

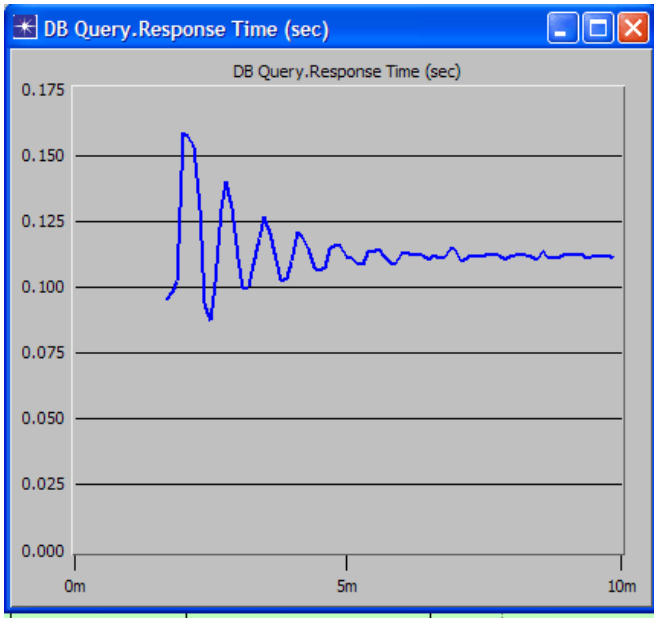


Figure 233: DB Query Response Time (sec)

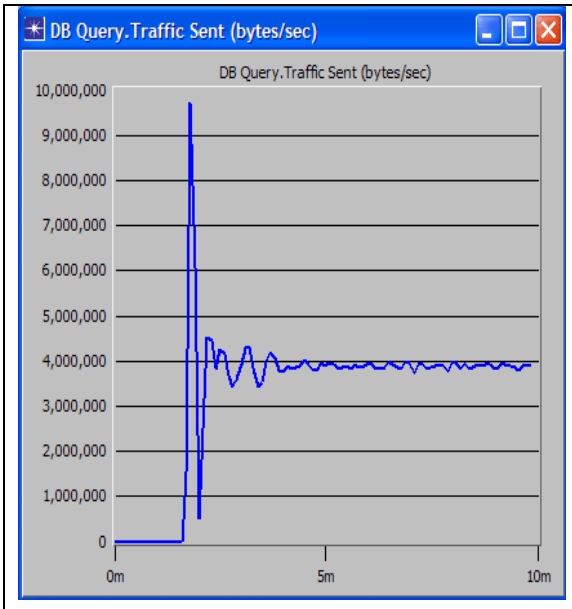


Figure 234: DB Query Traffic Sent (bytes/sec)

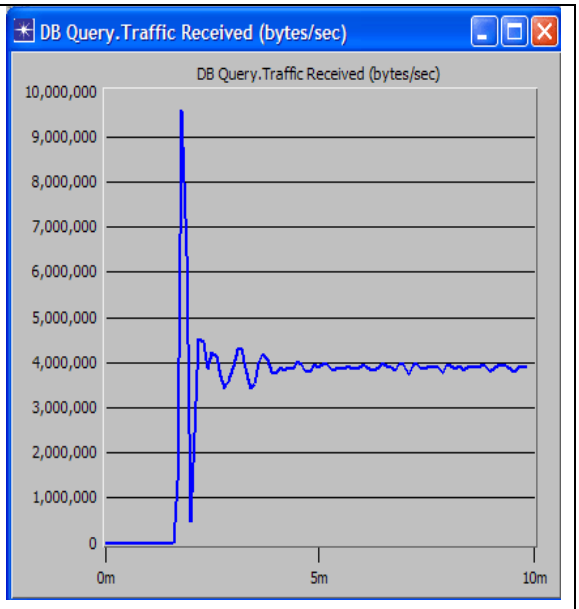


Figure 235: DB Query Traffic Received (bytes/sec)

Node	Application	Metric	Mean	St.Dev	95% CI
National	DB-Query	Response Time	0.11	0.01	[0.11, 0.12]
National	DB-Query	Traffic Sent	3.25M	1.66M	[2.92M, 3.57M]

Table 31: Global Statistics Values – DB Query

New York Node: DB Query

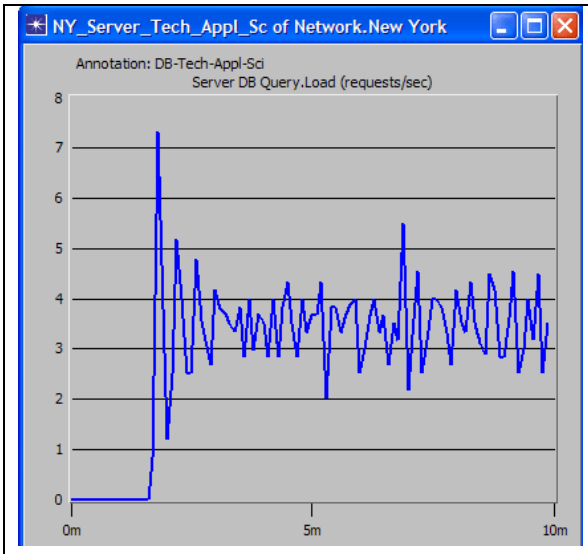


Figure 236: DB- Tech&Appli Load (requests/sec)

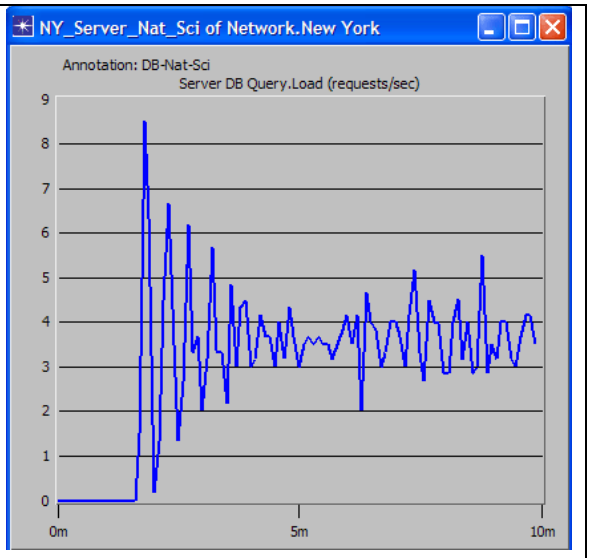


Figure 237: DB-NatSci&Math Load(requests/sec)

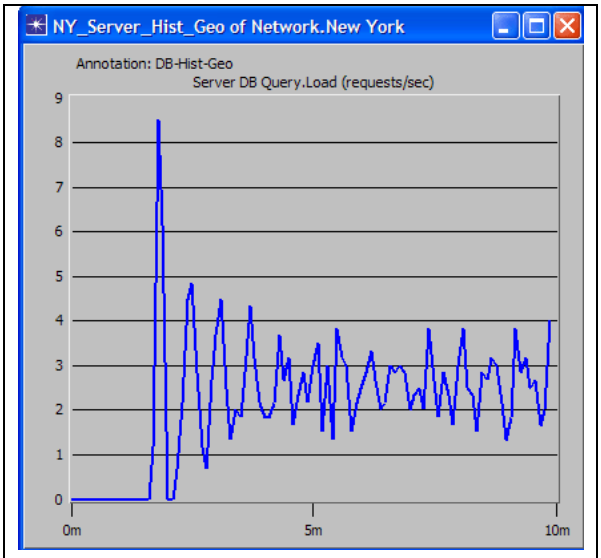


Figure 238: DB-Hist&Geo Load (requests/sec)

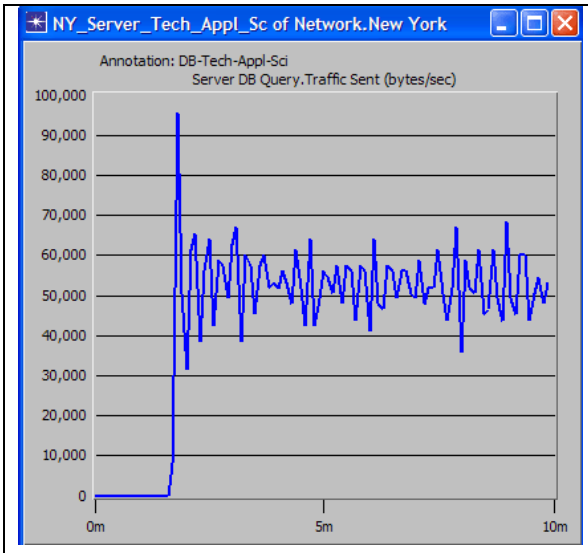


Figure 239: DB-Tech&Appli Traffic Sent (bytes/sec)

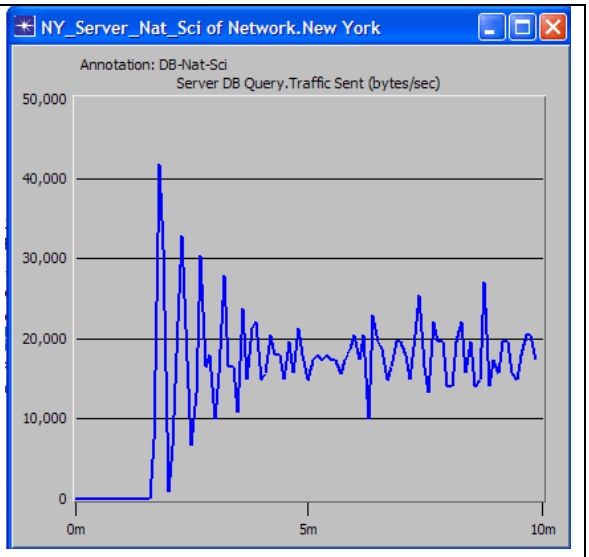


Figure 240: DB-NatSci&Math Traffic Sent (bytes/sec)

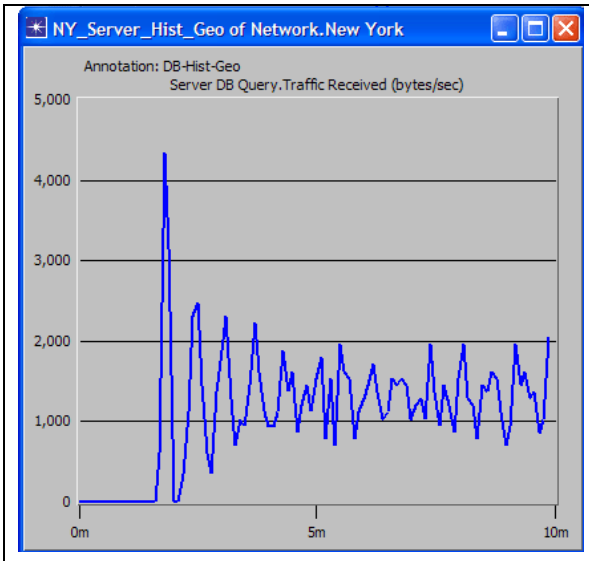


Figure 241: DB-Hist&Geo Traffic Received (bytes/sec)

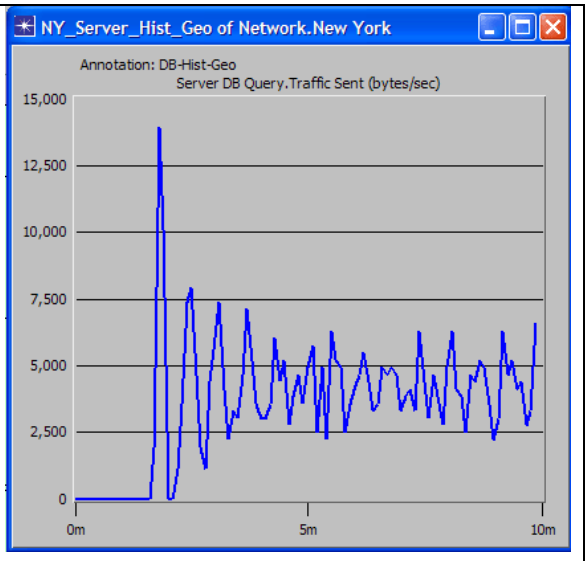


Figure 242: DB-Hist&Geo Traffic Sent (bytes/sec)

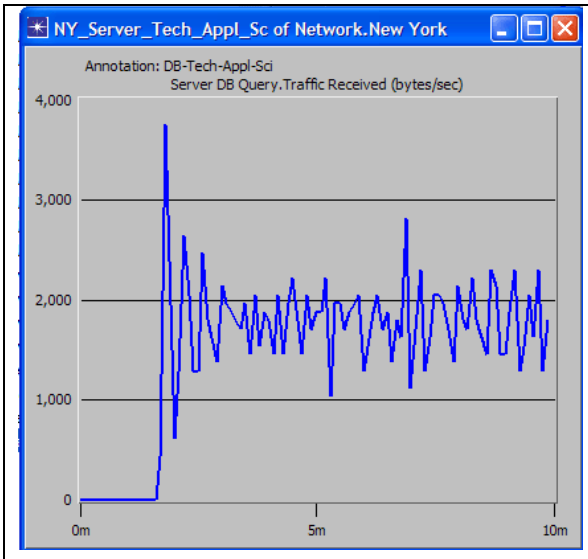


Figure 243: DB-Tech&Appli Traffic Recvd (bytes/sec)

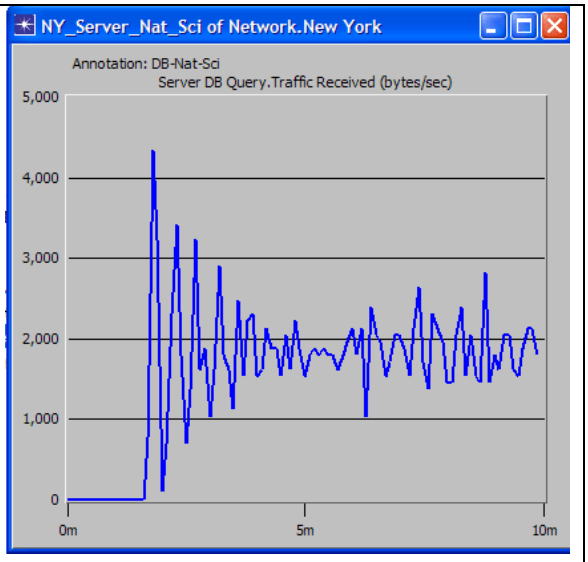


Figure 244: DB-NatSci&Math Traffic Recvd (bytes/sec)

Node	Server	Application	Metric	Mean	St.Dev	95%Conf.Intvl
NY	Tech&AppliSci	DB-Query	Load	2.90	1.55	[2.60, 3.21]
NY	Tech&AppliSci	DB-Query	Traffic Sent	43.83K	21.88K	[39.52K, 48.14K]
NY	Tech&AppliSci	DB-Query	Traffic Received	1.50K	791.20	[1.33K, 1.65K]
NY	NatSci&Math	DB-Query	Load	3.04	1.73	[2.70, 3.38]
NY	NatSci&Math	DB-Query	Traffic Sent	14.94K	8.51K	[13.26K, 16.62K]
NY	NatSci&Math	DB-Query	Traffic Received	1.56K	888.33	[1.38K, 1.73K]
NY	Hist&Geo	DB-Query	Load	2.18	1.47	[1.89, 2.47]
NY	Hist&Geo	DB-Query	Traffic Sent	3.58K	2.42K	[3.10K, 4.05K]
NY	Hist&Geo	DB-Query	Traffic Received	1.12K	755.06	[967.42, 1.27K]

Table 32: NY Node: DB-Query Statistics

LA Node: DB Query

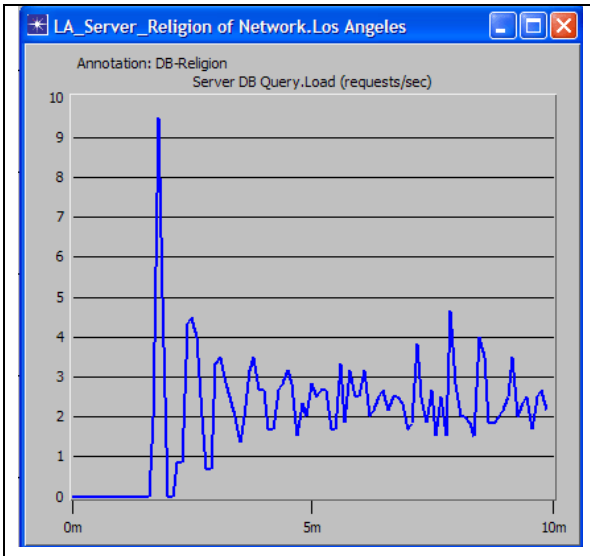


Figure 245: DB-Religion Load (requests/sec)

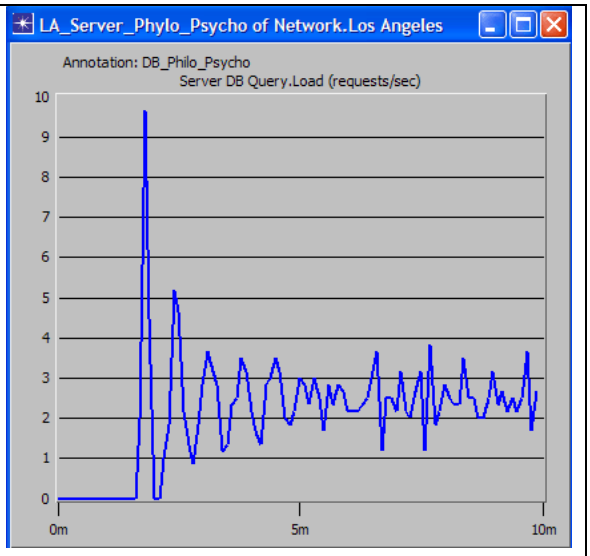


Figure 246: DB-Philo&Psycho Load (requests/sec)

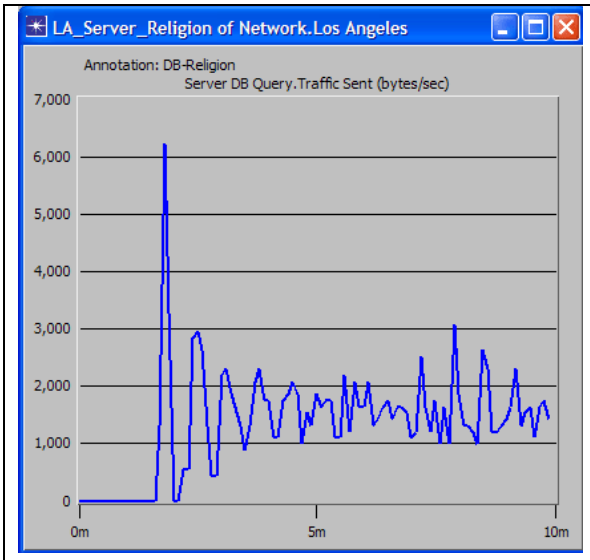


Figure 247: DB-Religion Traffic Sent (bytes/sec)

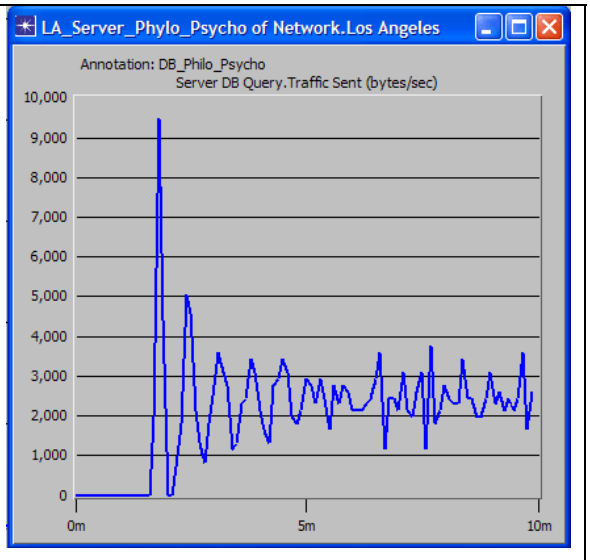


Figure 248: DB-Phi&Psy Traffic Sent (bytes/sec)

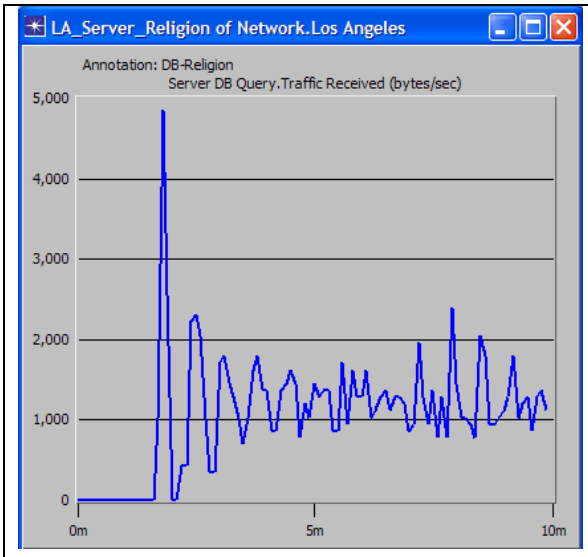


Figure 249: DB-Religion Traffic Recvd (bytes/sec)

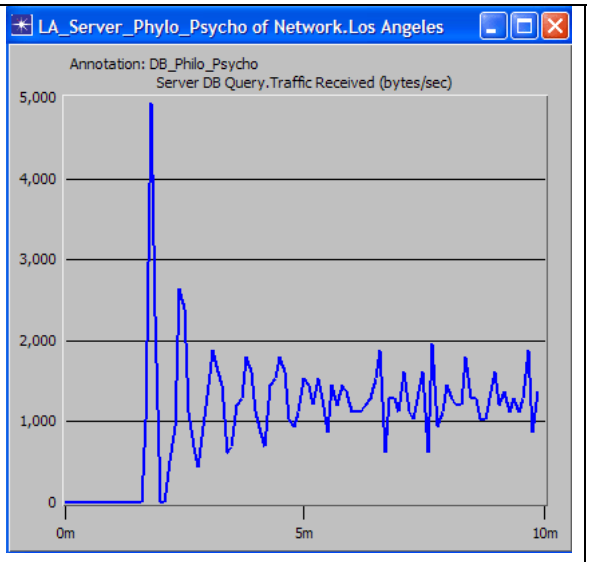


Figure 250: DB-Phi&Psy Traffic Recvd (bytes/sec)

Node	Server	Application	Metric	Mean	St.Dev	95%Conf.Intvl
LA	Religion	DB-Query	Load	2.06	1.44	[1.77, 2.34]
LA	Philo&Psycho	DB-Query	Load	2.10	1.43	[1.82, 2.38]
LA	Philo&Psycho	DB-Query	Traffic Sent	2.06K	1.40K	[1.78K, 2.34K]
LA	Philo&Psycho	DB-Query	Traffic Received	1.08K	732.07	[931.85, 1.22K]
LA	Religion	DB-Query	Traffic Sent	1.35K	946.38	[1.16K, 1.54K]

Table 33: LA Node: DB-Query Statistics

14.7.7 HTTP Statistics

Global Statistics – HTTP	
Metric	**Description**
Object Response Time (sec)	Specifies response time for each in-lined object from the HTML page.
Page Response Time (sec)	Specifies time required to retrieve the entire page with all the contained inline objects.
Traffic Sent (bytes/sec)	Average bytes per second submitted to the transport layer by all HTTP applications in the network.
Traffic Received (bytes/sec)	Average bytes per second forwarded to all the HTTP applications by the transport layer in the network.

Table 34: Global Statistics for HTTP

Node Statistics – HTTP	
Metric	**Description**
Load (request/sec)	Specifies response time for each in-lined object from the HTML page.
Traffic Sent (bytes/sec)	Average bytes per second submitted to the transport layer by all HTTP applications in the network.
Traffic Received (bytes/sec)	Average bytes per second forwarded to all the HTTP applications by the transport layer in the network.

Table 35: Node Statistics for HTTP

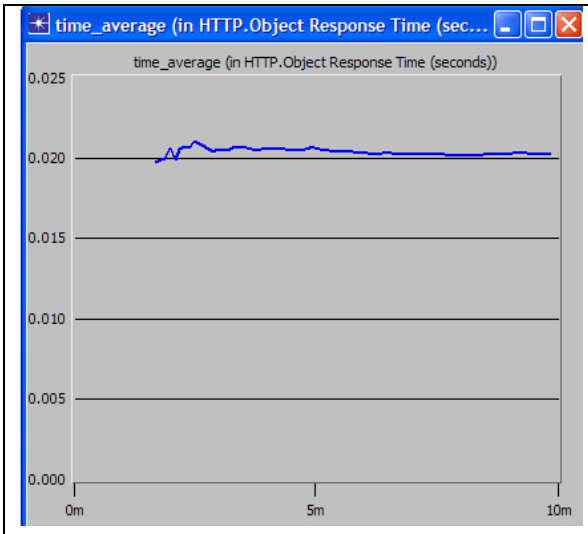


Figure 251: INCN HTTP Object Response Time (sec)

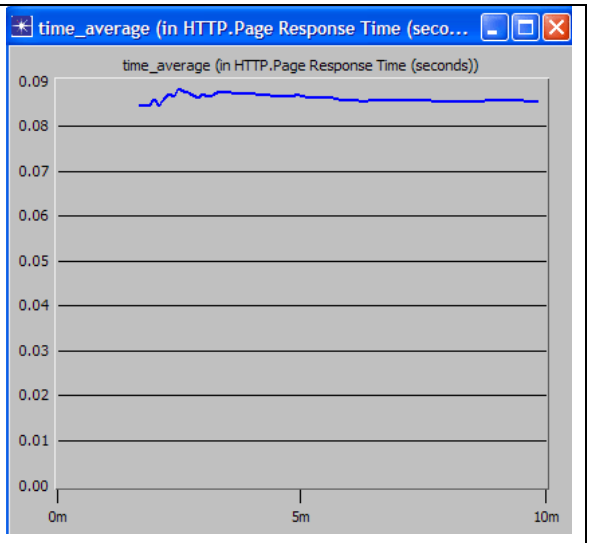


Figure 252: INCN HTTP Page Response Time (sec)

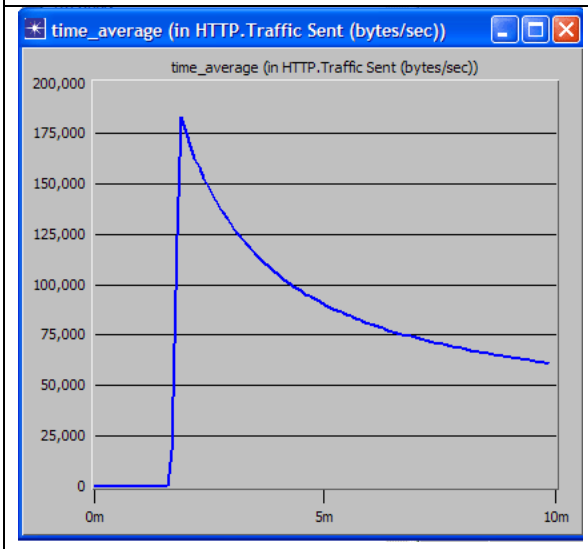


Figure 253: INCN HTTP Traffic Sent (bytes/sec)

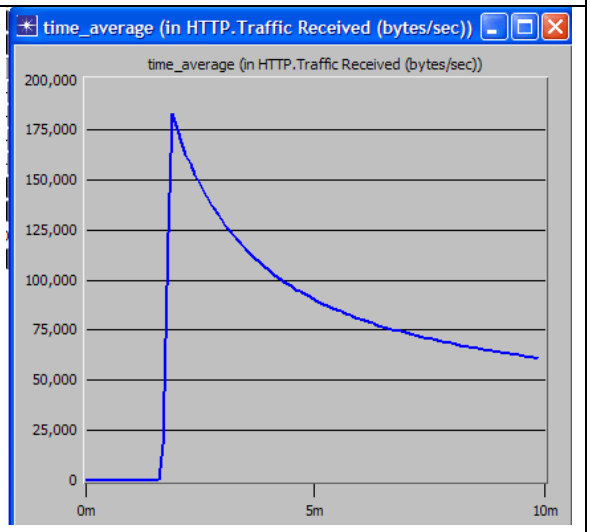


Figure 254: INCN HTTP Traffic Received (bytes/sec)

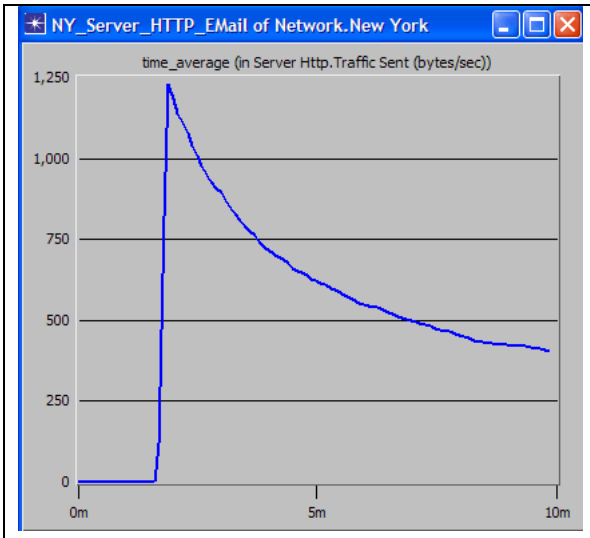


Figure 255: INCN HTTP Traffic Sent NY (bytes/sec)

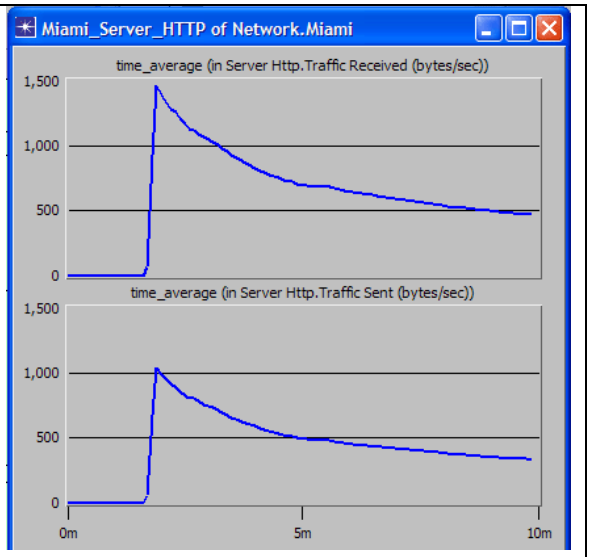


Figure 256: INCN HTTP Traffic Sent Miami (bytes/sec)

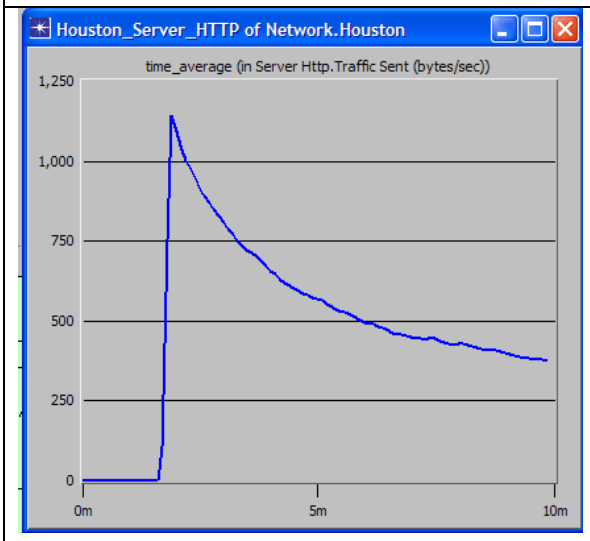


Figure 257: INCN HTTP Traffic Sent Houston (bytes/sec)

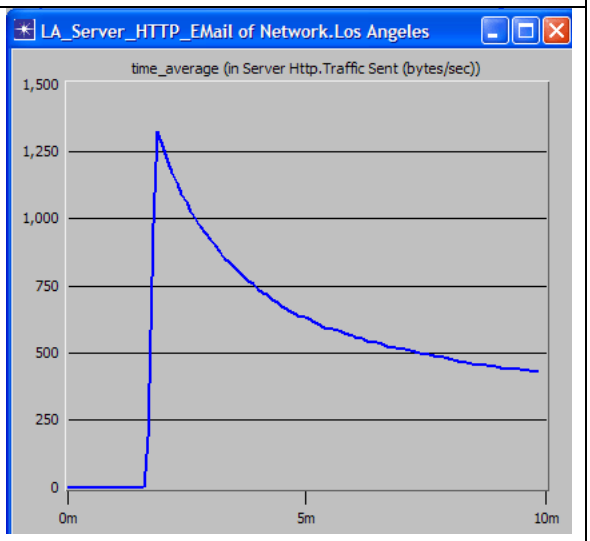


Figure 258: INCN HTTP Traffic Sent LA (bytes/sec)

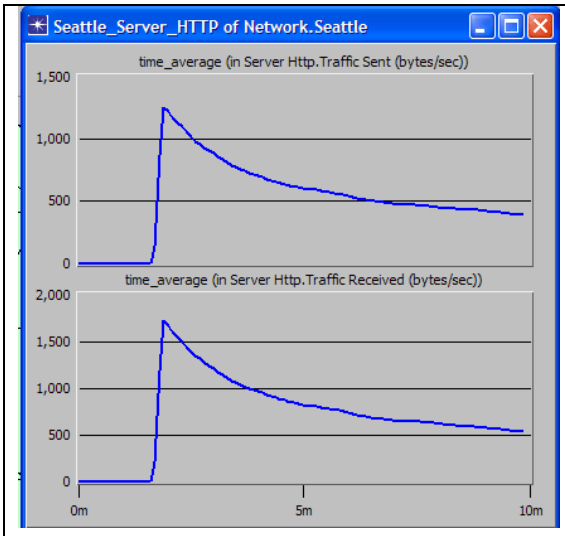


Figure 259: HTTP Traffic Sent/Recvd Seattle

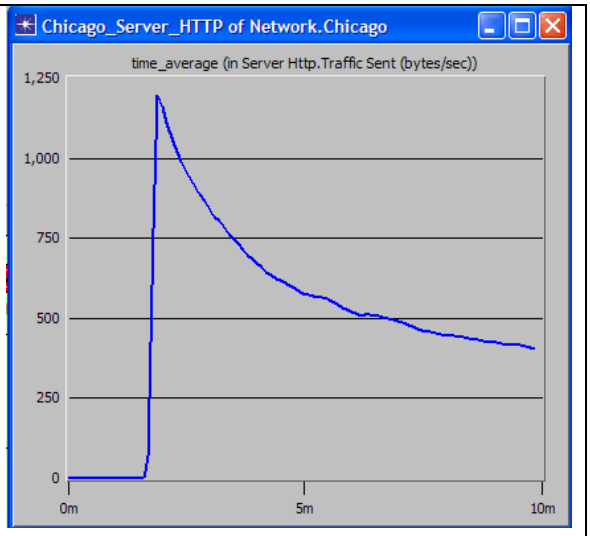


Figure 260: HTTP Traffic Sent Chicago (bytes/sec)

Node	Server	Application	Metric	Mean	St.Dev	95%Conf.Intvl
National		HTTP	Object Resp.Time	0.02	0.0002	[0.02, 0.02]
National		HTTP	Page Resp.Time	0.08	0.0008	[0.08, 0.08]
National		HTTP	Traffic Sent	76.19K	44.85K	[67.35K, 85.02K]
National		HTTP	Traffic Received	76.18K	44.85K	[67.35K, 85.02K]
NY	HTTP_EMAIL	HTTP	Load	1.62	6.53	[0.33, 2.90]
NY	HTTP_EMAIL	HTTP	Traffic Sent	517.00	306.50	[456.61, 577.36]
NY	HTTP_EMAIL	HTTP	Traffic Received	736.72	442.14	[649.62, 823.82]
LA	HTTP_EMAIL	HTTP	Traffic Sent	538.82	321.23	[475.54, 602.10]
Chicago	HTTP	HTTP	Traffic Sent	496.19	291.81	[438.70, 553.67]
Chicago	HTTP	HTTP	Traffic Received	537.25	2.18K	[107.32, 967.18]
Seattle	HTTP	HTTP	Traffic Sent	510.87	307.88	[450.22, 571.51]
Seattle	HTTP	HTTP	Traffic Received	697.92	421.07	[614.98, 780.87]
Miami	HTTP	HTTP	Traffic Received	603.06	358.84	[532.37, 673.75]
Miami	HTTP	HTTP	Traffic Sent	427.30	254.50	[377.16, 477.43]
Houston	HTTP	HTTP	Traffic Sent	473.95	279.44	[418.91, 529.00]
Houston	HTTP	HTTP	Traffic Received	525.00	2.10K	[110.82, 939.18]

Table 36: Values of Global & Node Statistics: HTTP

14.7.8 EMail Statistics

Global Statistics – E-Mail	
Metric	**Description**
Download Response Time (sec)	Time elapsed between sending request for emails and receiving emails from email server in the network. This time includes signaling delay for the connection setup.
Upload Response Time (sec)	Time elapsed between sending emails to the email server and receiving acknowledgements from the email server. This time includes signaling delay for the connection setup.
Traffic Sent (bytes/sec)	Average bytes per second submitted to the transport layers by all email applications in the network.
Traffic Received (bytes/sec)	Average bytes per second forwarded to all the E-Mail applications by the transport layer in the network.

Table 37: Global Statistics for E-Mail

Node Statistics – E-Mail	
Metric	**Description**
Load (requests/sec)	Rate at which any Email requests arrive at the server. Note that these requests could belong to different sessions maintained at the server. Also note that requests for the same session is queued until the first request is completed.
Traffic Sent (bytes/sec)	Average bytes per second submitted to the transport layer by the email application in this node.
Traffic Received (bytes/sec)	Average bytes per second forwarded to the email application by the transport layer in this node.

Table 38: Node Statistics for E-Mail

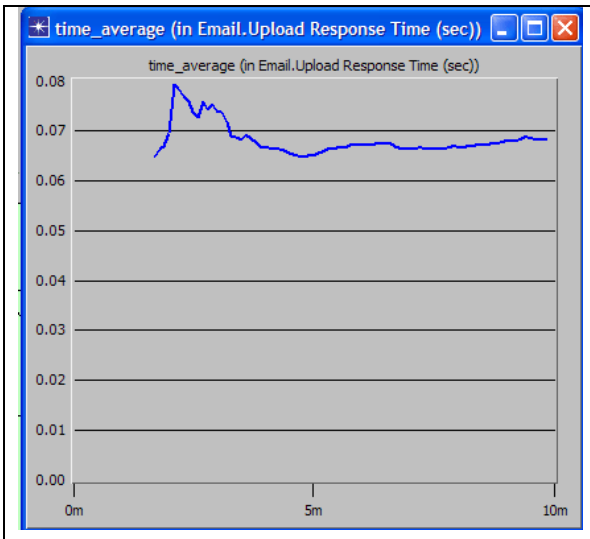


Figure 261: INCN E-Mail Upload Response Time (sec)

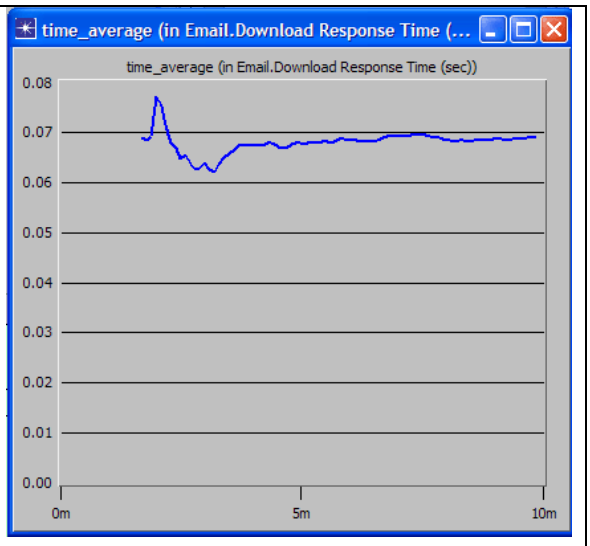


Figure 262: INCN E-Mail Download Response Time (sec)

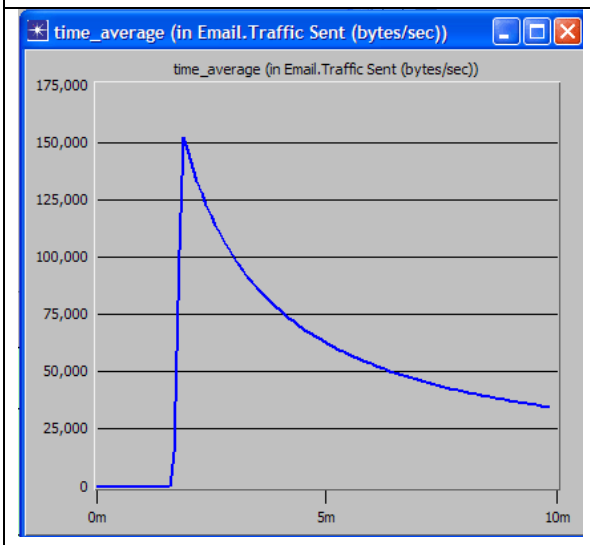


Figure 263: INCN E-Mail Traffic Sent (bytes/sec)

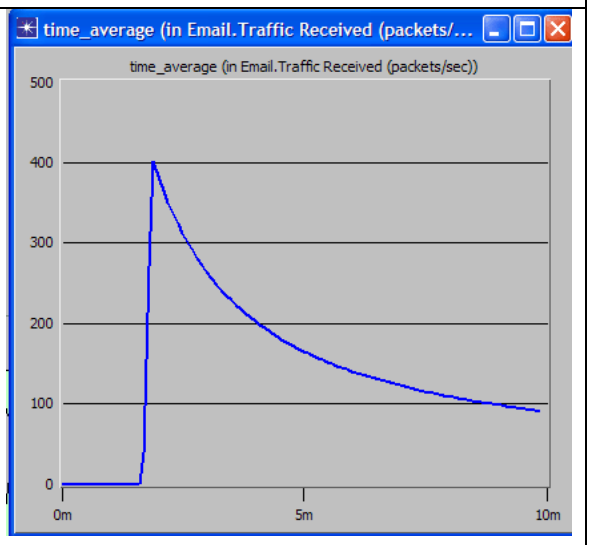


Figure 264: INCN E-Mail Traffic Received (packets/sec)

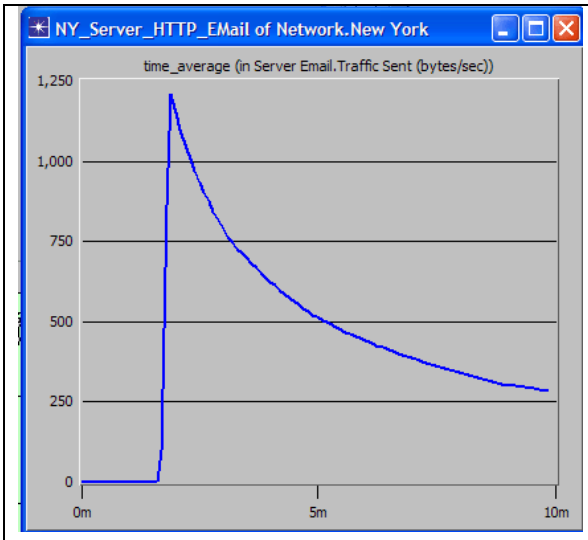


Figure 265: INCN E-Mail Traffic Sent NY (bytes/sec)

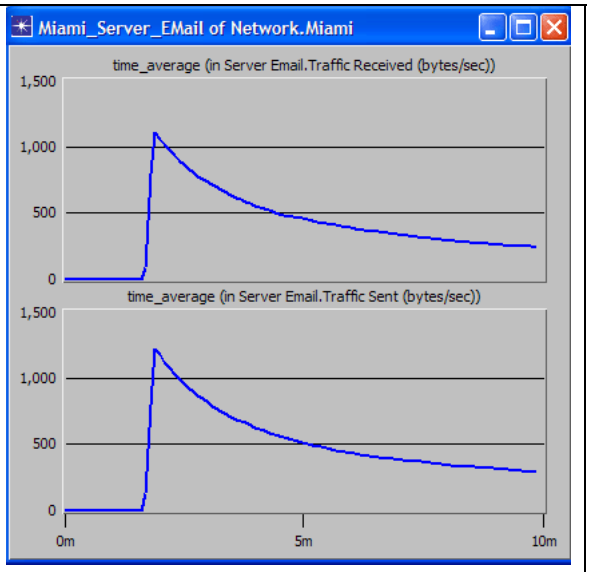


Figure 266: INCN E-Mail Traffic Sent/Received Miami (bytes/sec)

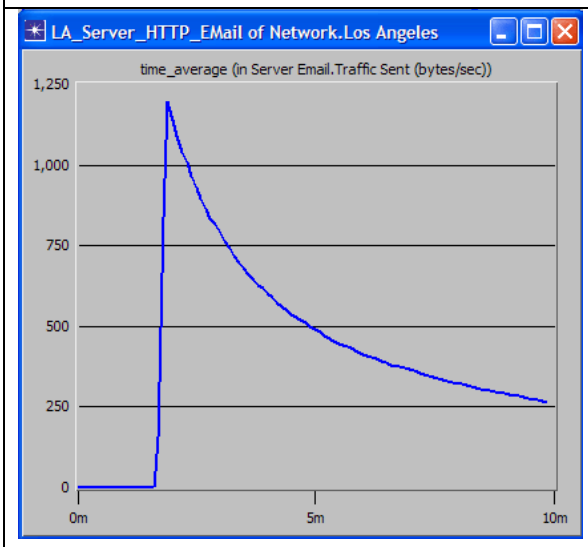


Figure 267: INCN E-Mail Traffic Sent LA (bytes/sec)

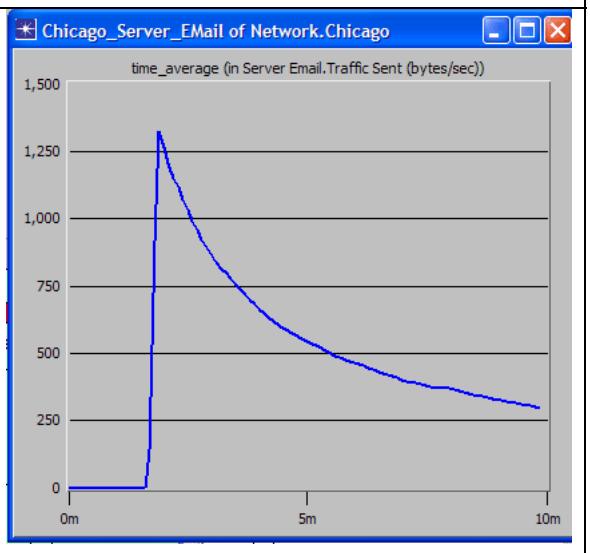


Figure 268: INCN E-Mail Traffic Sent Chicago (bytes/sec)

Node	Server	Application	Metric	Mean	St.Dev	95%Conf.Intvl
National		E-MAIL	Upload Response Time	0.07	0.003	[0.07, 0.07]
National		E-MAIL	Download Response Time	0.07	0.002	[0.07, 0.07]
National		E-MAIL	Traffic Sent	53.50K	36.42K	[46.31K, 60.66K]
National		E-MAIL	Traffic Received	141.11	96.10	[122.18, 160.04]
NY	HTTP_EMAIL	E-MAIL	Traffic Sent	433.60	290.17	[376.42, 490.74]
LA	HTTP_EMAIL	E-MAIL	Traffic Sent	419.13	287.06	[362.59, 475.68]
Chicago	EMAIL	E-MAIL	Traffic Sent	465.63	316.60	[403.27, 528.00]
Chicago	EMAIL	E-MAIL	Traffic Received	280.65	1.61K	[-36.30, 598.00]
Seattle	EMAIL	E-MAIL	Traffic Sent	372.91	258.95	[321.90, 423.92]
Seattle	EMAIL	E-MAIL	Traffic Received	344.98	234.97	[298.70, 391.27]
Miami	EMAIL	E-MAIL	Traffic Received	387.31	268.78	[334.36, 440.26]
Miami	EMAIL	E-Mail	Traffic Sent	437.00	294.30	[379.02, 494.96]
Houston	EMAIL	E-MAIL	Traffic Sent	248.23	1.40K	[-26.00, 522.43]
Houston	EMAIL	E-MAIL	Traffic Received	305.11	1.82K	[-52.85, 663.08]

Table 39: Values of Global & Node Statistics: E-Mail

14.7.9 TCP Statistics

Global Statistics – TCP	
Metric	**Description**
Delay (sec)	Delay of packets received by the TCP layers in the complete network, for all connections. It is measured from the time an application data packet is sent from the source TCP layer to the time it is completely received by the TCP layer in the destination node.

Table 40: Global Statistics for TCP

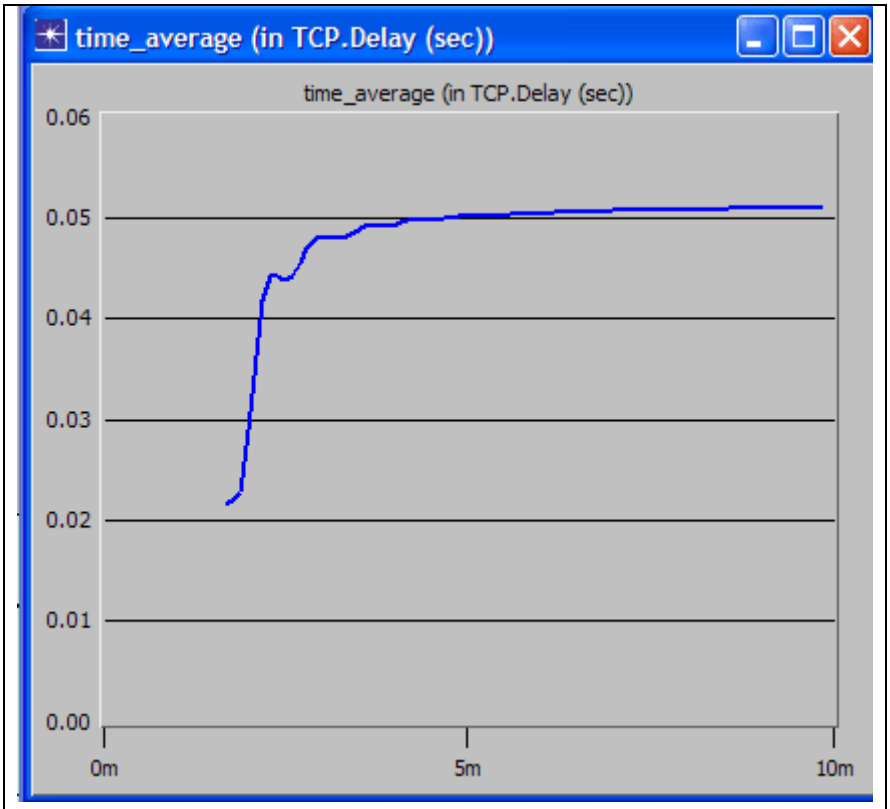


Figure 269: INCN TCP Delay (sec)

Node	Application	Metric	Mean	St.Dev	95%Conf.Intvl
National	TCP	Delay (sec)	0.05	0.006	[0.05, 0.05]

Table 41: Values of Global Statistics: TCP Delay (sec)

14.7.10 IP Statistics

Global Statistics – IP	
Metric	**Description**
Traffic Dropped (packets/sec)	Number of IP datagrams dropped by ALL nodes in the network across all IP interfaces. Possible reasons for dropping an IP datagram: (i) Insufficient space in the central processor's queue. (ii) Insufficient space in a slot processor's buffer (when slot is enabled). (iii) Maximum number of hops exceeded by an IP datagram. (iv) A local router interface was not found to be used as the next hop. (v) On routing nodes, the route table failed to yield route to destination.

Table 42: Global Statistics for IP

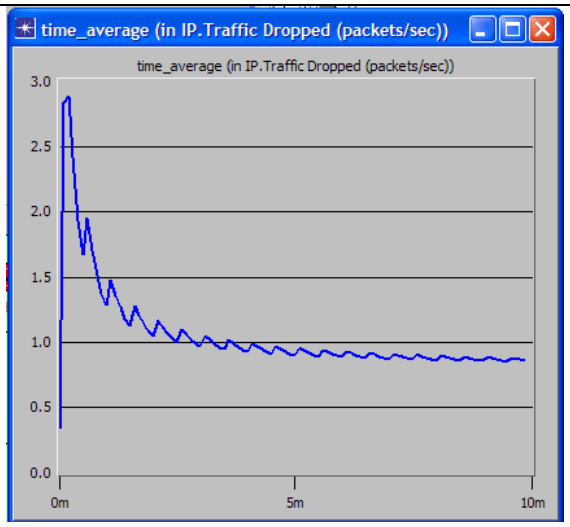


Figure 270: INCN IP Traffic Dropped (packets/sec)

Node	Application	Metric	Mean	St.Dev	95%Conf.Intvl
National	IP	Traffic Dropped (packets/sec)	1.05	0.37	[0.98, 1.13]

Table 43: Values of Global Statistic: IP Traffic Dropped (packets/sec)

14.7.11 Ethernet Statistics

Global Statistics – Ethernet	
Metric	**Description**
Delay (sec)	This statistic represents the end to end delay of all packets received by all the stations

Table 44: Global Statistics for Ethernet

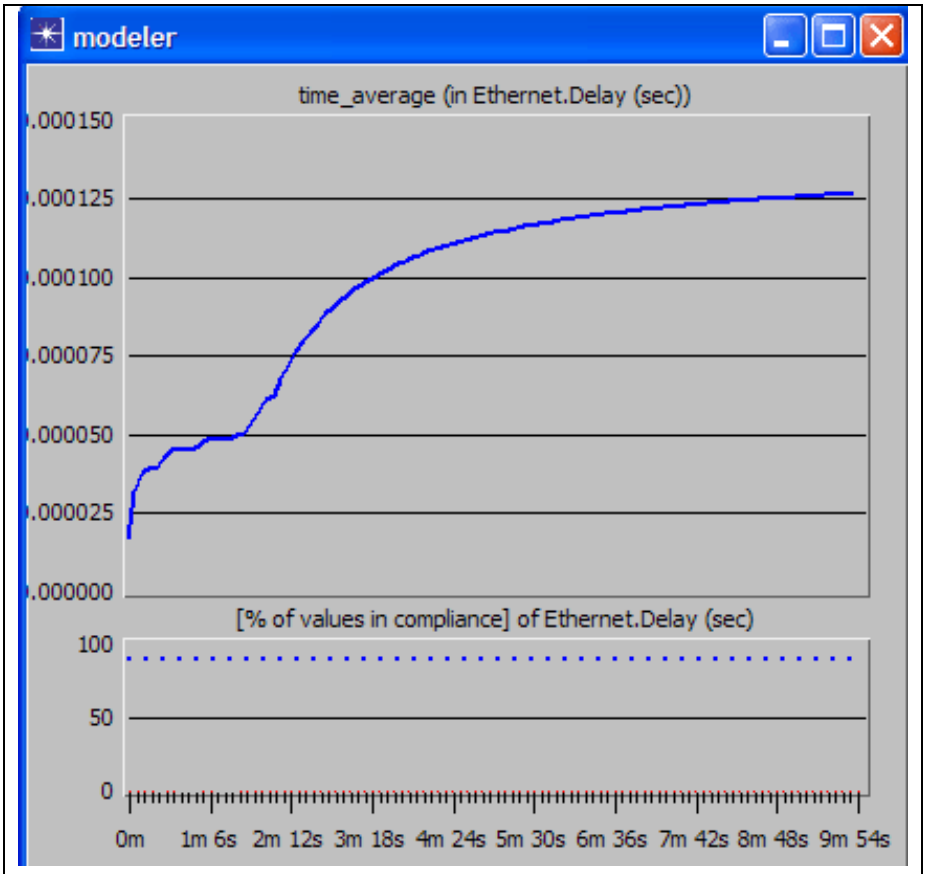


Figure 271: INCN Ethernet Delay (sec)

Node	Application	Metric	Mean	St.Dev	95%Conf.Intvl
National	Ethernet	Delay (sec)	0.00009	0.00003	[0.00009, 0.0001]

Table 45: Values Global Statistics: Ethernet Delay (sec)

14.7.12 Link Statistics

Link Statistics – Port-to-Port	
Metric	**Description**
Utilization (%)	This statistic represents the percentage of the consumption to date of an available channel bandwidth, where a value of 100.0 would indicate full usage.
Throughput (bits/sec)	This statistic represents the average number of bits successfully received or transmitted by the receiver or transmitter channel per unit time, in bits per second.
Queuing Delay (sec)	This statistic represents instantaneous measurements of packet waiting times in the transmitter channel's queue. Measurements are taken from the time a packet enters the transmitter channel queue to the time the last bit of the packet is transmitted.

Table 46: Link (Port-to-Port) Statistics

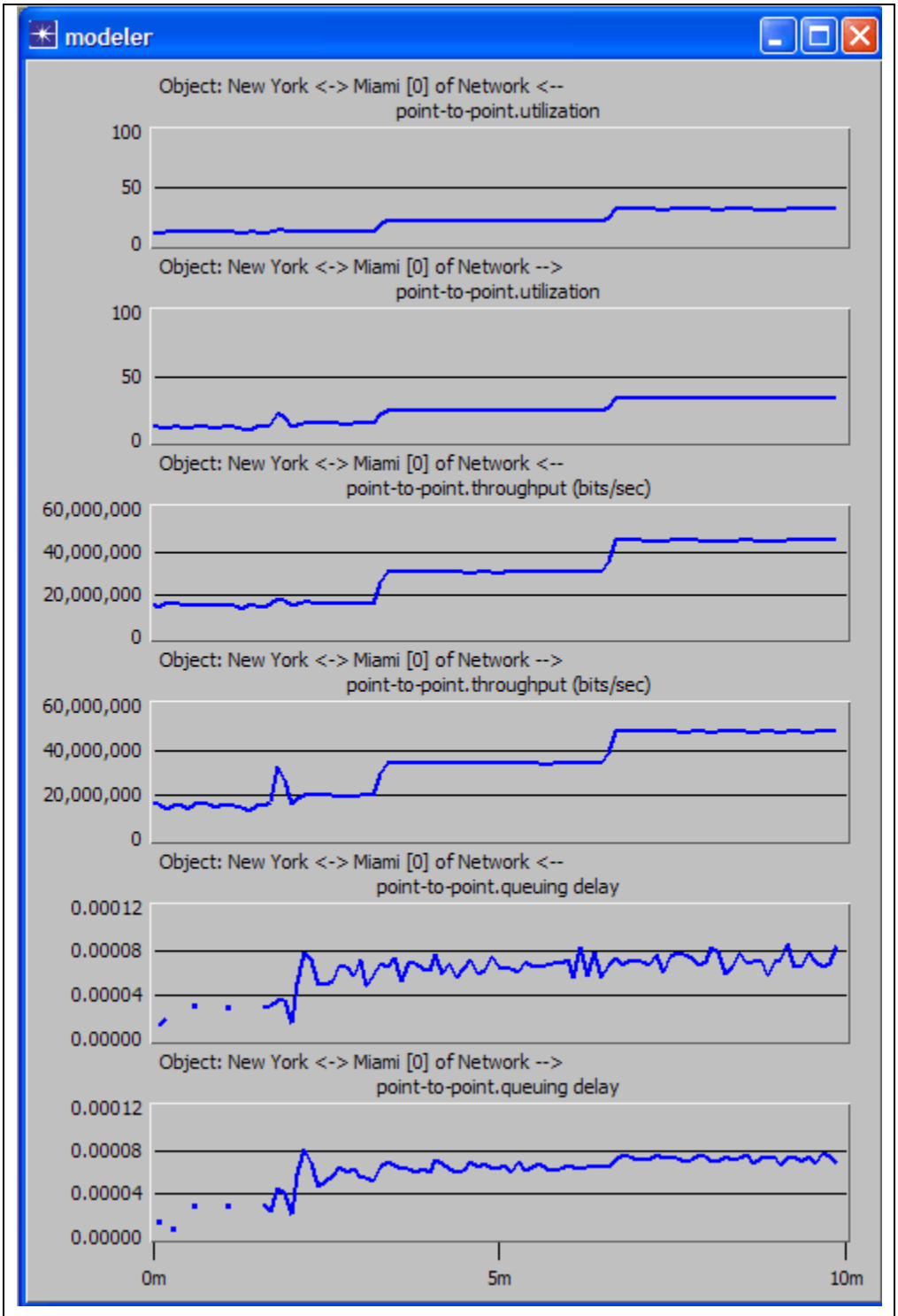


Figure 272: INCN Link Statistics New York – Miami

Node	Metric	Mean	St.Dev	95%Conf.Intvl
NY-Mia	Utilization(%) ←	20.40	8.19	[18.78, 22.00]
NY-Mia	Utilization(%) →	22.32	8.61	[20.62, 24.01]
NY-Mia	Utilization (%)	42.72		
NY-Mia	Throughput (bits/sec) ←	30.30M	12.17M	[27.90, 32.70]
NY-Mia	Throughput (bits/sec) →	33.16M	12.80	[30.64, 35.68]
NY-Mia	Throughput (bits/sec)	63.46M		
NY-Mia	Queuing Delay (sec) ←	0.00006	1.52	[6.01, 6.65]E-05
NY-Mia	Queuing Delay (sec) →	0.00006	1.42	[6.02, 6.62]E-05
NY-Mia	Queuing Delay (sec)	0.00012		

Table 47: Link Statistics: New York - Miami

Link: OC-3 (155 Mbps), and Utilization: 43%, then Throughput: 43% x 155 Mbps = 66.65 Mbps

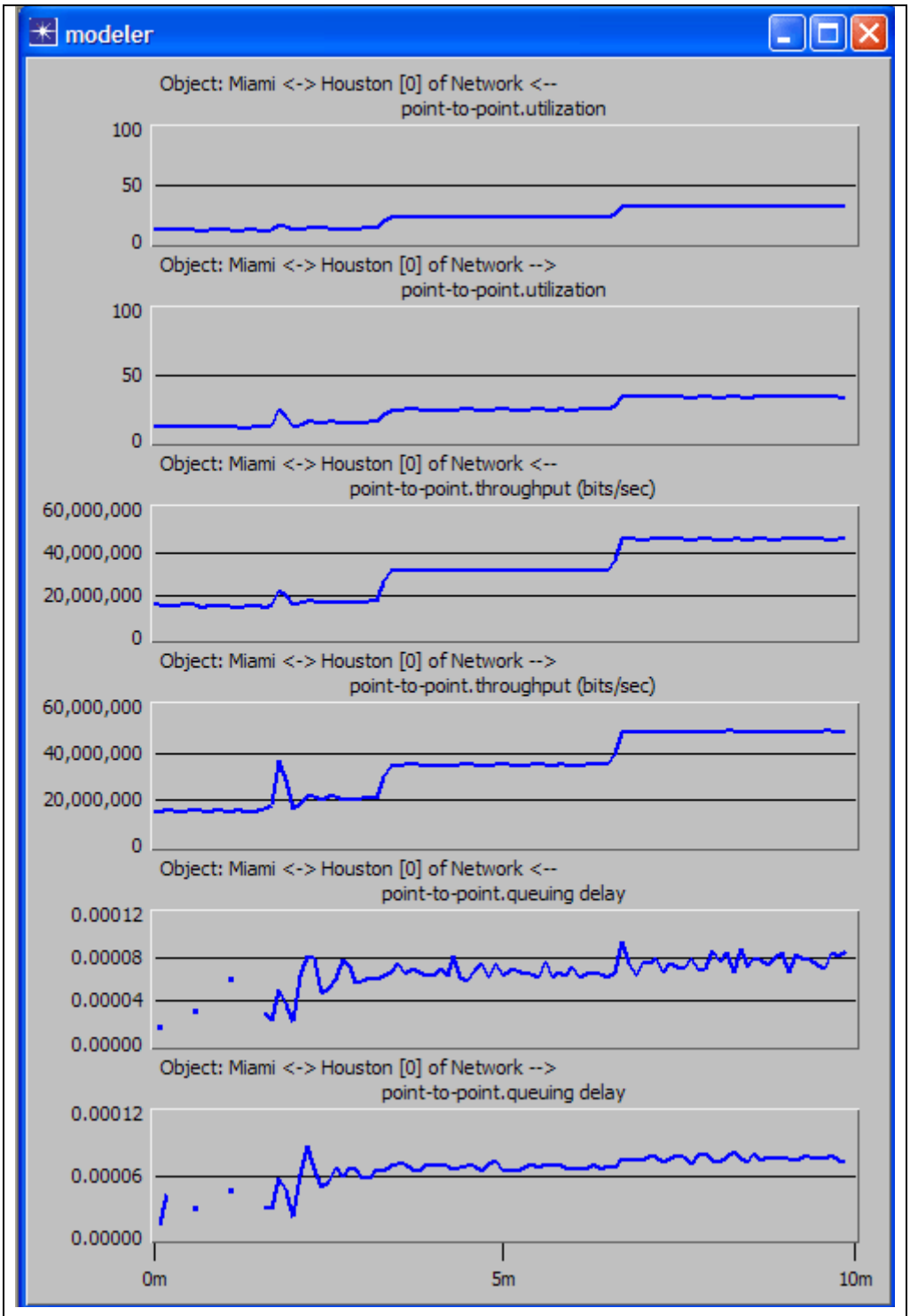


Figure 273: INCN Link Statistics Miami – Houston

Node	Metric	Mean	St.Dev	95%Conf.Intvl
Mia-Hou	Utilization(%) ←	20.90	8.30	[19.27, 22.53]
Mia-Hou	Utilization(%) →	22.82	8.67	[21.11, 24.53]
Mia-Hou	Utilization (%)	43.72		
Mia-Hou	Throughput (bits/sec) ←	31.06M	12.33M	[28.63M, 33.49]
Mia-Hou	Throughput (bits/sec) →	34.00M	12.90	[31.37, 36.49]
Mia-Hou	Throughput (bits/sec)	65.06M		
Mia-Hou	Queuing Delay (sec) ←	0.00007	1.44	[6.39, 7.00]E-05
Mia-Hou	Queuing Delay (sec) →	0.00007	1.35	[6.45, 7.01]E-05
Mia-Hou	Queuing Delay (sec)	0.00014		

Table 48: Link Statistics: Miami - Houston

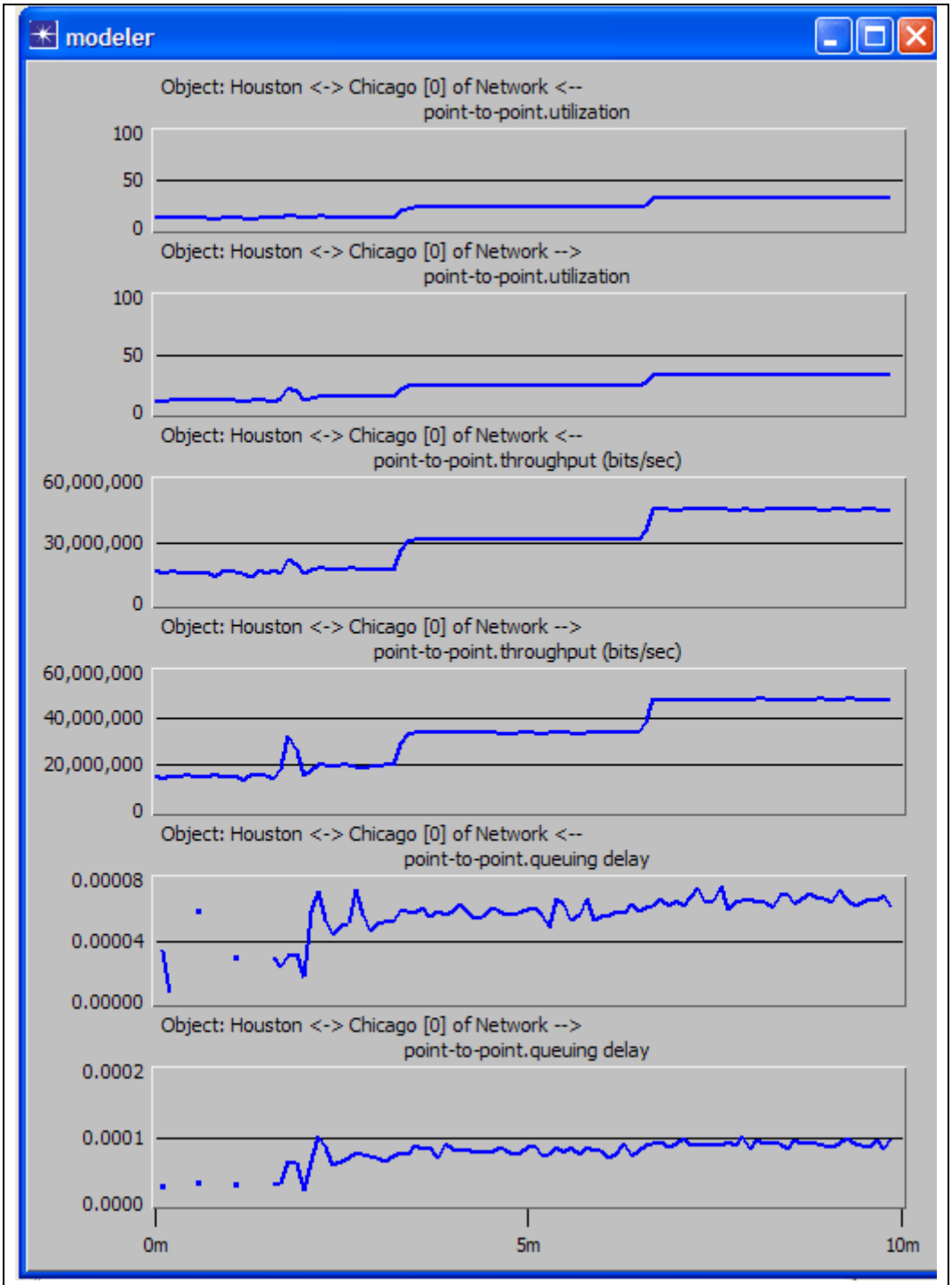


Figure 274: INCN Link Statistics Houston – Chicago

Node	Metric	Mean	St.Dev	95%Conf.Intvl
Hou-Chi	Utilization(%) ←	20.92	8.30	[19.30, 22.55]
Hou-Chi	Utilization(%) →	22.30	8.60	[20.60, 24.00]
Hou-Chi	Utilization (%)	43.22		
Hou-Chi	Throughput (bits/sec) ←	31.09M	12.30M	[28.67M, 33.51M]
Hou-Chi	Throughput (bits/sec) →	33.13M	12.76M	[30.61M, 35.64M]
Hou-Chi	Throughput (bits/sec)	64.22M		
Hou-Chi	Queuing Delay (sec) ←	0.00006	1.27	[0.00006, 0.00006]
Hou-Chi	Queuing Delay (sec) →	0.00008	1.69	[0.00007, 0.00008]
Hou-Chi	Queuing Delay (sec)	0.00014		

Table 49: Link Statistics: Houston - Chicago

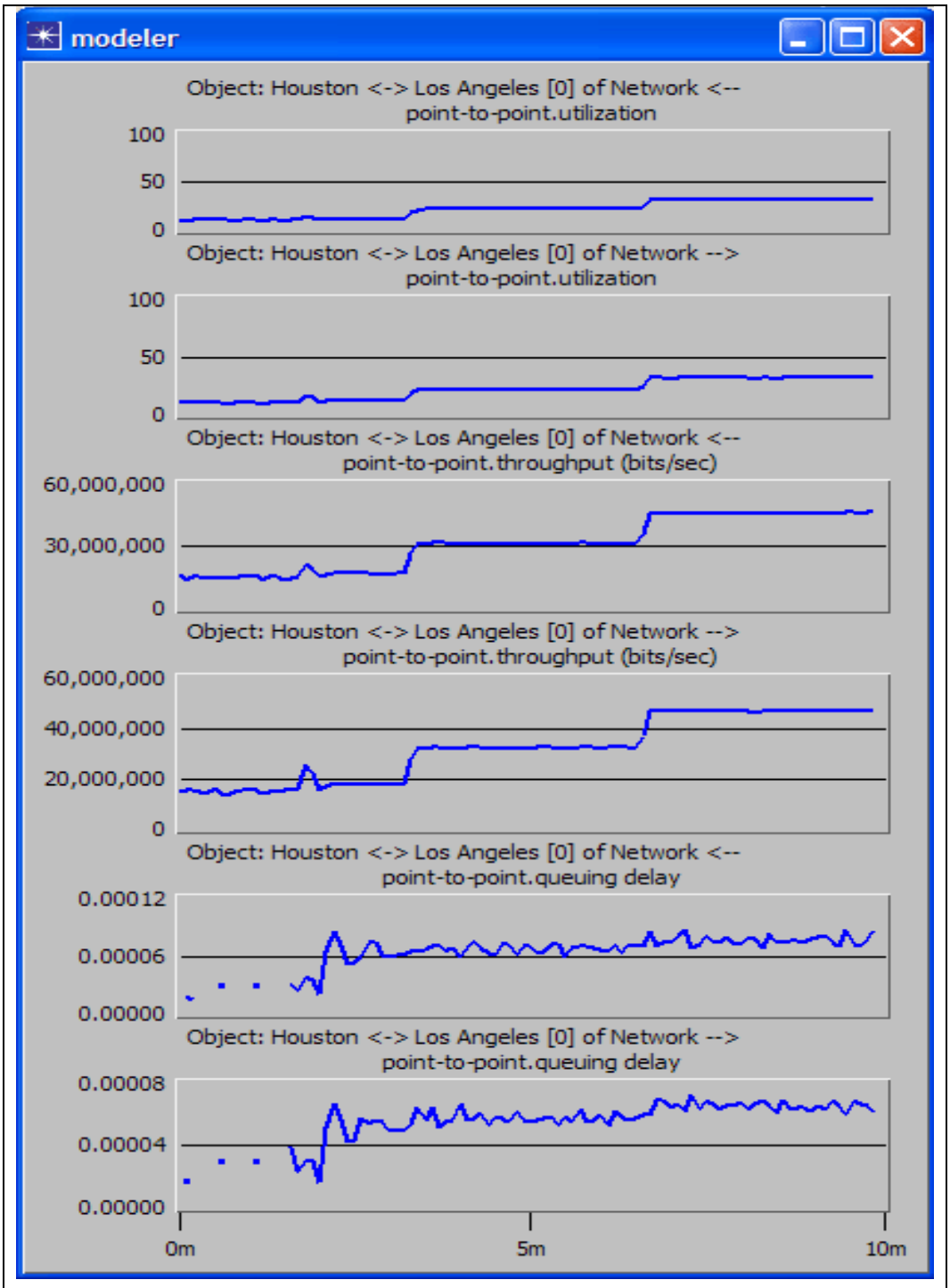


Figure 275: INCN Link Statistics Houston – Los Angeles

Node	Metric	Mean	St.Dev	95%Conf.Intvl
Hou-LA	Utilization(%) ←	20.85	8.29	[19.22, 22.48]
Hou-LA	Utilization(%) →	21.40	8.38	[19.74, 23.04]
Hou-LA	Utilization (%)	42.25		
Hou-LA	Throughput (bits/sec) ←	31.00 M	12.31M	[28.56M, 33.41M]
Hou-LA	Throughput (bits/sec) →	31.80M	12.46M	[29.33M, 34.24M]
Hou-LA	Throughput (bits/sec)	62.80M		
Hou-LA	Queuing Delay (sec) ←	000006	1.55	[6.31, 6.96]E-05
Hou-LA	Queuing Delay (sec) →	0.00006	1.15	[5.46, 5.94]E-05
Hou-LA	Queuing Delay (sec)	0.00012		

Table 50: Link Statistics: Houston – Los Angeles

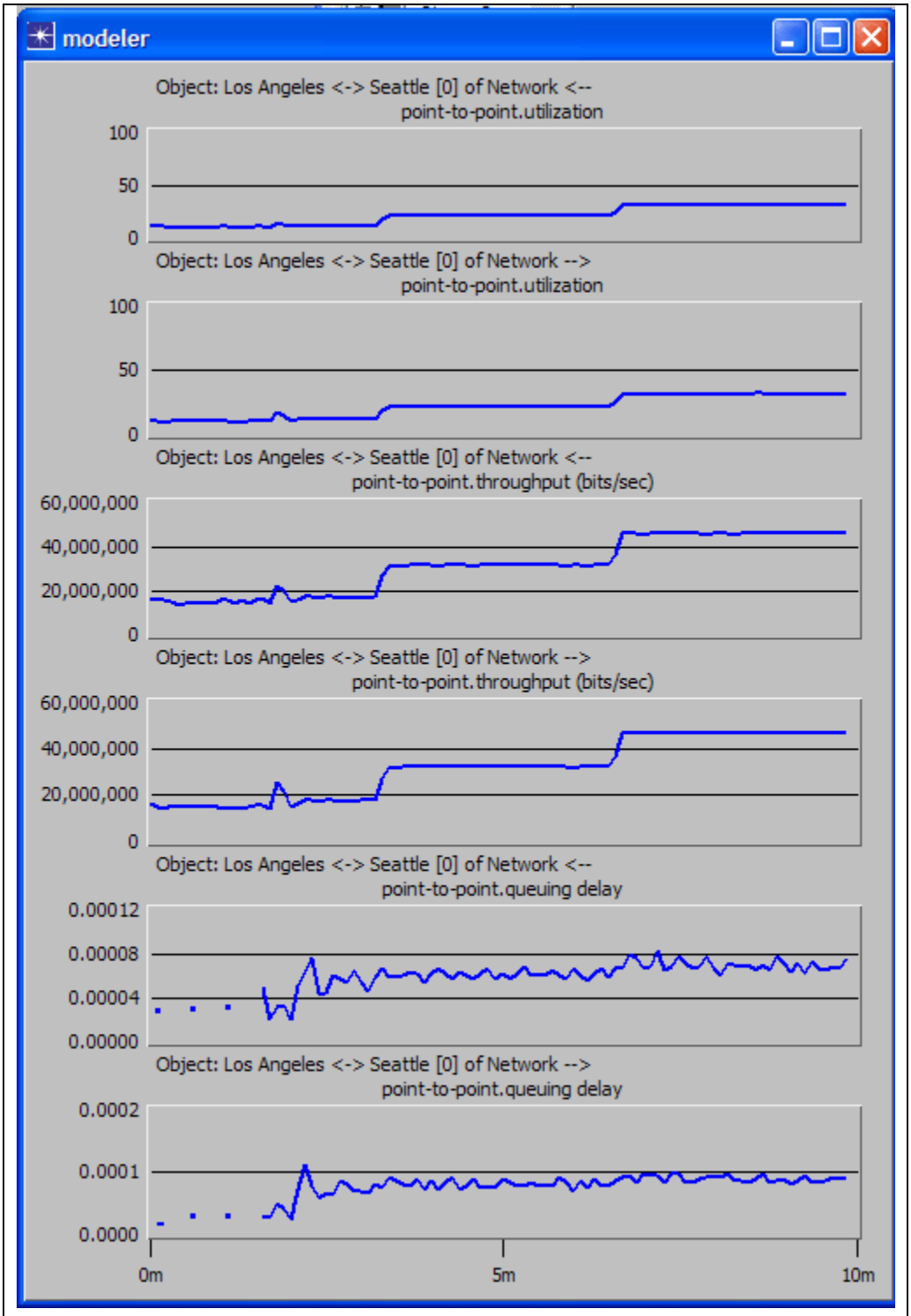


Figure 276: INCN Link Statistics Los Angeles – Seattle

Node	Metric	Mean	St.Dev	95%Conf.Intvl
LA-Sea	Utilization(%) ←	20.90	8.25	[19.27, 22.52]
LA-Sea	Utilization(%) →	21.38	8.35	[19.73, 23.02]
LA-Sea	Utilization (%)	42.28		
LA-Sea	Throughput (bits/sec) ←	31.06M	12.26M	[28.64M, 33.47M]
LA-Sea	Throughput (bits/sec) →	31.77M	12.40M	[29.32M, 34.21M]
LA-Sea	Throughput (bits/sec)	62.83M		
LA-Sea	Queuing Delay (sec) ←	0.00006	1.25E-05	[5.81, 6.33]E-05
LA-Sea	Queuing Delay (sec) →	0.00008	1.90E-05	[7.40, 8.20]E-05
LA-Sea	Queuing Delay (sec)	0.00014		

Table 51: Link Statistics Los Angeles – Seattle

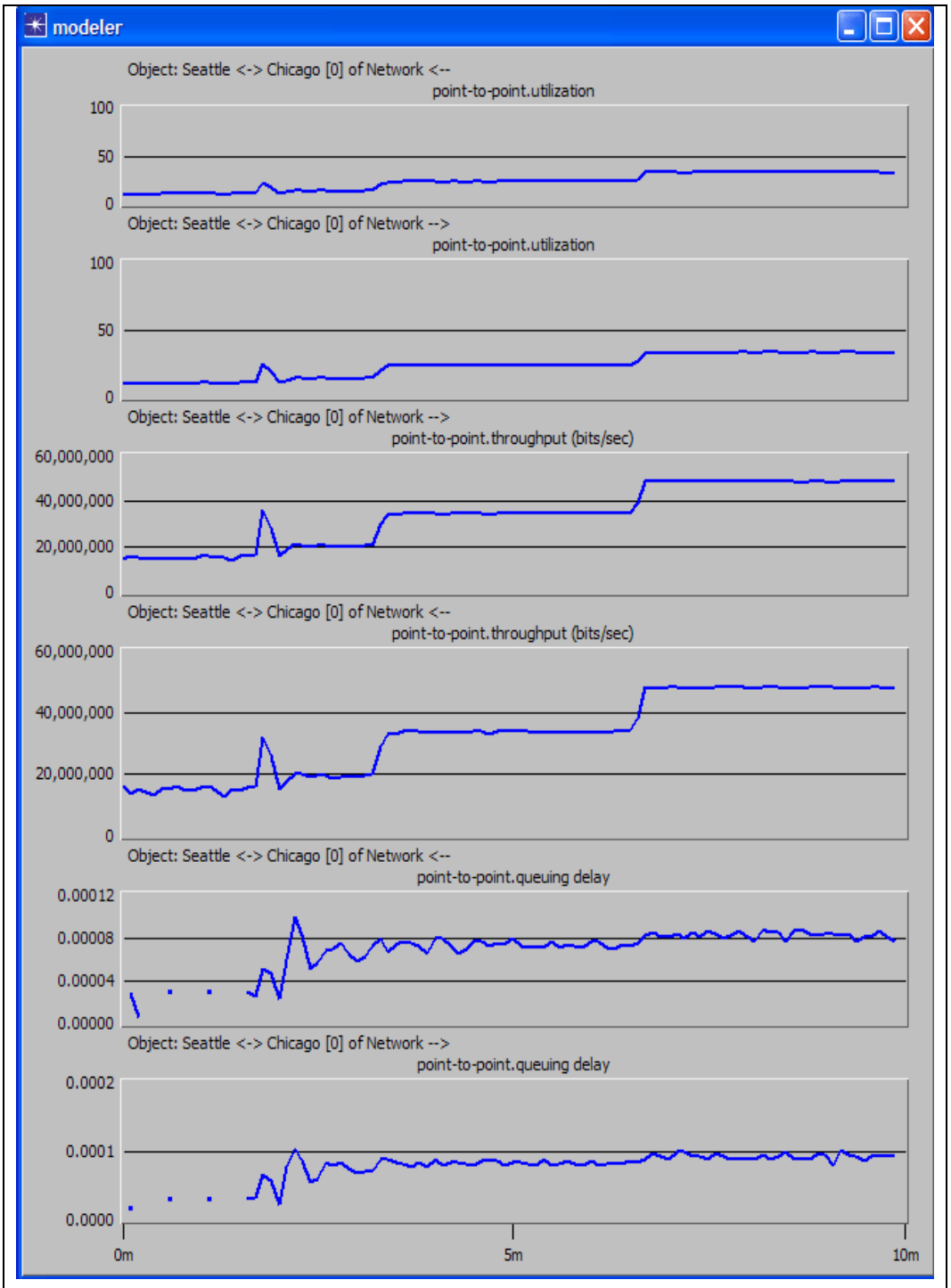


Figure 277: INCN Link Statistics Seattle – Chicago

Node	Metric	Mean	St.Dev	95%Conf.Intvl
Sea-Chi	Utilization(%) ←	22.27	8.56	[20.58, 23.96]
Sea-Chi	Utilization(%) →	22.78	8.68	[21.07, 24.50]
Sea-Chi	Utilization (%)	45.05		
Sea-Chi	Throughput (bits/sec) →	33.86M	12.90M	[31.32M, 36.40M]
Sea-Chi	Throughput (bits/sec) ←	33.09M	12.73M	[30.60M, 35.60M]
Sea-Chi	Throughput (bits/sec)	66.95M		
Sea-Chi	Queuing Delay (sec) ←	0.00007	0.00002	[0.00007, 0.00008]
Sea-Chi	Queuing Delay (sec) →	0.00008	0.00002	[0.00008, 0.00008]
Sea-Chi	Queuing Delay (sec)	0.00015		

Table 52: Link Statistics Seattle - Chicago

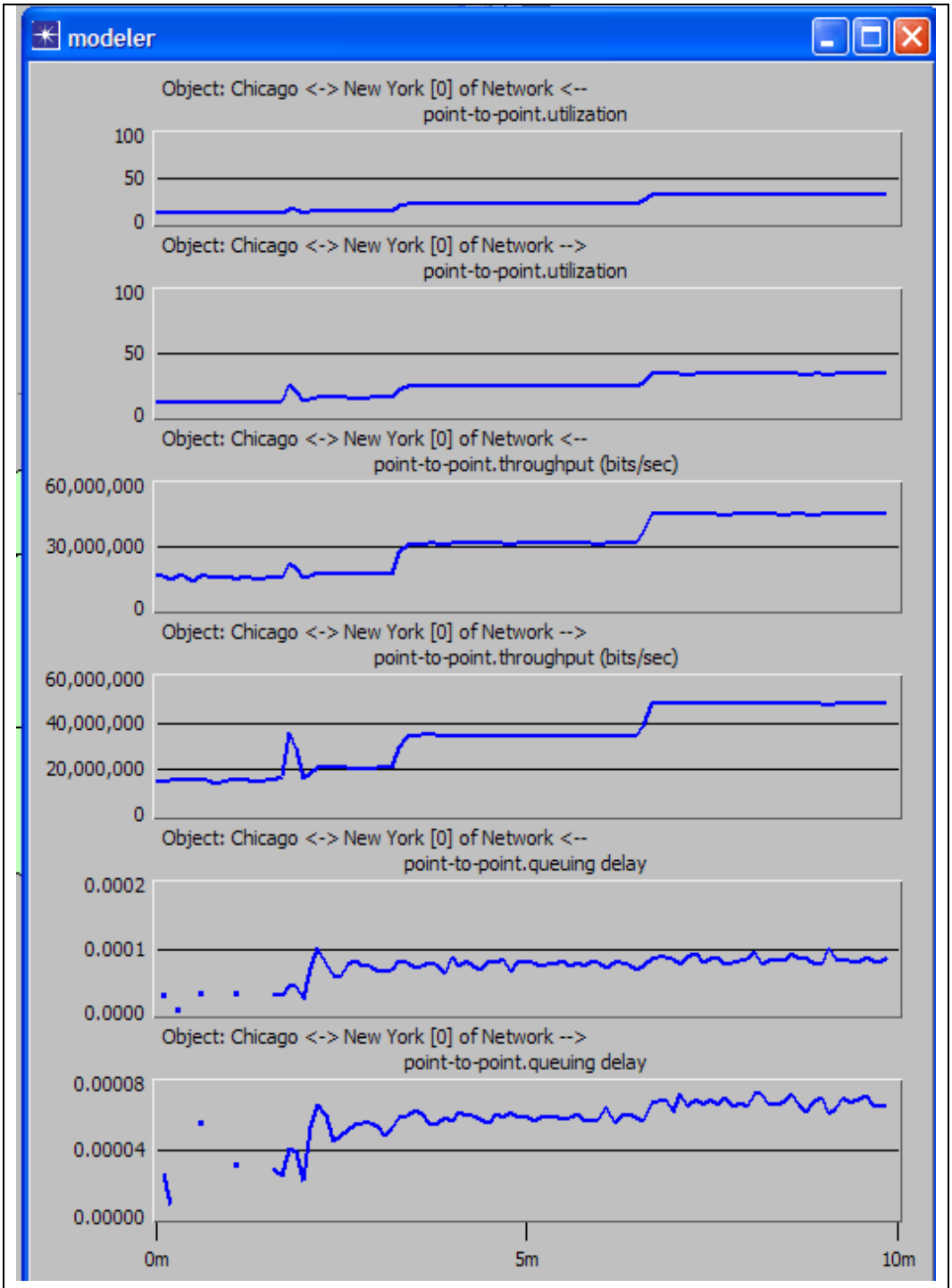


Figure 278: INCN Link Statistics Chicago – New York

Node	Metric	Mean	St.Dev	95%Conf.Intvl
Chi-NY	Utilization(%) ←	21.00	8.20	[19.40, 22.62]
Chi-NY	Utilization(%) →	22.82	8.70	[21.11, 24.54]
Chi-NY	Utilization (%)	43.82		
Chi-NY	Throughput (bits/sec) ←	31.20M	12.26M	[28.78M, 33.61M]
Chi-NY	Throughput (bits/sec) →	33.92M	12.92M	[31.37M, 36.46M]
Chi-NY	Throughput (bits/sec)	65.12M		
Chi-NY	Queuing Delay (sec) ←	0.00007	0.00002	[0.00007, 0.00008]
Chi-NY	Queuing Delay (sec) →	0.00006	0.00001	[0.00006, 0.00006]
Chi-NY	Queuing Delay (sec)	0.00013		

Table 53: Link Statistics Chicago – New York

14.7.13 Analysis of the Simulation Results

The main focus of our analysis is on the traffic generated by the Database queries that are expected to be treated intelligently by our network. Each query, depending on their knowledge domain according to the Dewey classification scheme, is to be routed properly to the correct knowledge base.

Each one of our subnets, see figure 227, at a particular location is composed of 10 LANs and each of these LANs is the source of data traffic. We are allowing them to generate Web browsing traffic, eMail traffic, and database queries traffic of five specific domains: Religion, History & Geography, Philosophy & Psychiatry, Natural Science & Mathematics, and Technology & Applied Sciences. We also placed knowledge bases in New York and Los Angeles. In New York we have 3 knowledge bases for Technology & Applied Sciences, Natural Science & Mathematics, and History & Geography; and 2 knowledge bases in Los Angeles: Religion and Philosophy & Psychology. These applications are set up as being generated simultaneously. However, at any given time, only a particular type of application is being processed at the router.

Figures 279 and 280 show the amount of traffic that is being sent to the NY server 'DB-Tech-Appl-Sci' and the amount of database query traffic of the type 'Technical & Applied Sciences' generated by a particular LAN of our subnet in the New York node respectively. The graphs show the same amount of traffic which then proves that our SCP is doing its job of selecting the correct knowledge base after identifying the type of packet is dealing with.

It should be noted that the same results will yield for any of the ten LANs and any of the five knowledge domain we used in our simulation run

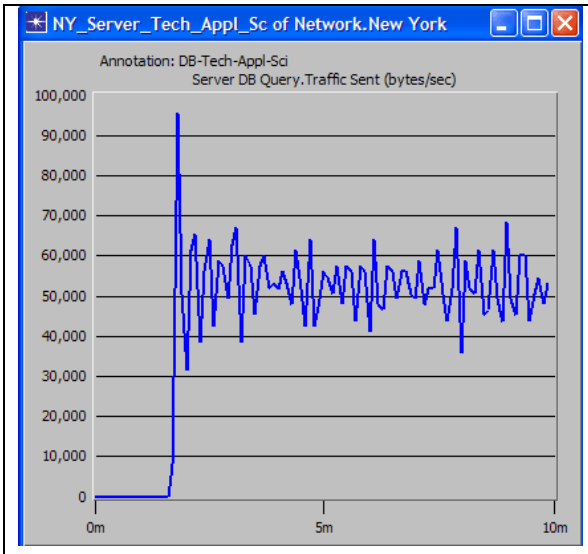


Figure 279: DB-Tech-Appl NY Server Traffic Sent (bytes/sec)

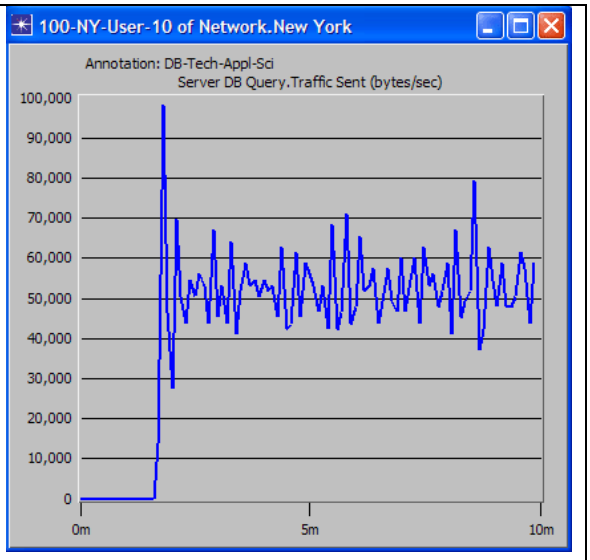


Figure 280: DB-Tech-Appl NY LAN-10 Traffic Sent (bytes/sec)

Each workstation of a LAN (100 users) transmits 8,192 bytes every 17 seconds (interarrival time) of database queries related to Technology & Applied Sciences topic. This traffic corresponds to 48,000 bytes/sec which is comparatively close to what figures 278 and 279 are showing.

A common feature of the traffic for DB-queries is the oscillations that appear right after the warm-up time (between 100-110 ms). These oscillations are due to the rapid routing of different packets belonging to different knowledge domains. The router is receiving packets from different domains every second and it has to switch them to the appropriate knowledge base. It takes a certain amount of time (>10min) for the oscillations to settle down.

Obviously, our routing algorithm, see figure 231, will fail when the identification of the packet is not done properly. However to further demonstrate the integrity of our systems, the simulation shows low rates of packets being dropped – it's less than a packet throughout the simulation run.

We have five database applications running at each of the six 1,000-user nodes of the network. Using the percentage values shown in Table 27, the calculated average DB-Query traffic for each DB application is:

. Philo&Psycho: 16.4 bytes/sec

. Religion: 8.0 bytes/sec

. Hist&Geo 16.4 bytes/sec

. NatSci&Math 182.0 bytes/sec

. Tech&AppliSci 481.8 bytes/sec

. Total: 704.6 bytes/sec

Consequently, the 6,000 users generate a total DB-Query traffic of 4.2 M bytes/sec which is exactly what figures 234 and 235 shows.

The response time for DB-Query traffic is averaged at 110ms well within acceptable values. Human factors studies provided the basis for a target of sub-second response time, $T < 1.0s$ [16]. Lower value for interarrival time of database queries would not have affected response times.

Queuing delay is the delay in the queue, and it is a function of the amount of traffic entering the router and the size of the queue. Queuing, as its name suggests, is a way of lining up network traffic so that it flows according to the prioritization preferences – in our case, WFQ/RED. The number of packets in a queue increases exponentially as utilization on an output port increases. Queuing is set up at the edge-routers for campus networks because LAN switches generally do not understand queuing methods. Our experiment resulted in a average of 75μsec queuing delay which is an acceptable value [42, 47, 50].

Utilization is the percent of available bandwidth in use. The average utilization for any of the 7 links of our network is 43%. We introduced background noise traffic accounting for up to 30% of the traffic. This background traffic is employed for two reasons: first to 'mimic' the daily behavior of traffic starting early in the morning, going up to noon, slowing down right after noon, and going back up early afternoon, before dying down towards the end of the business day, around 5pm, and second, to accommodate for noise traffic which is inevitably present to a certain degree in any actual network traffic. Therefore, we are underutilizing the link capacity of 155Mbps by around 40% corresponding to about 20% background and 65Mbps throughput.

Throughput measures the amount of link bandwidth is being used by the network traffic. In our network, the overall traffic intensity corresponds to a flow of 63 millions of bits per second at any point of the circuit.

15. Conclusion and Future Research

The thesis examines issues related to performance modeling, management, and control of high-speed campus networks. It also expands the network by adding IN capability to it in order to provide a specific service, i.e., that of processing knowledge queries of an intelligent education systems (IES). This service is different from that provided by an intelligent database system (IDB) because it is a network environment system relying in a signaling network and the function of a knowledge control point that switches network traffic based on the content domain and not based on node address of the packets. Whereas IDB is computer-centric, IES with its KCP is network-centric.

The intent of the thesis is firstly to develop insight into the operations and performance of enterprise networks, and secondly to present an intelligent educational system (IES) based on a conceptual model of IENA (Intelligent Educational Network Architecture) which functions as a 'negotiator' between knowledge seekers and knowledge providers. The first part of the thesis consisted in configuring the national (NCN) and international campus networks (ICN) with different links, traffic loads, and quality of service protocols. The resulting simulation statistics were analyzed in order to understand how the networks react with different scenarios and to serve as a basis for suggesting a method for performance modeling of campus networks.

The efficiency of a network link can be measured by link utilization which is defined as the bandwidth of a link in use divided by the total bandwidth of the link. The possibility of full utilization in the network under a given traffic load is an important criterion in network design. Our analysis showed that T-3 can not be the link for NCN and ICN because it can not handle the amount of traffic being generated by the network users.

Without any background, OC-3 can be employed in the NCN, but it is not advisable to use it in the ICN because it utilizes close to half of the link's capacity connecting two of the subnet pairs. Either OC-12 or OC-48 can be the link of choice for NCN and ICN. However, none of the links can be used in our campus networks when background traffic is present at the levels of above 30% of the link capacity. These levels result in extreme high utilizations.

Simulation with hybrid traffic, which consists of explicit traffic (from the applications) and background traffic (according to degradation levels), is particularly important because background traffic models are a fundamental component of packet-level network simulators since the network load drives the packet drop rate, queuing delay, end-to-end delay variation, and available network throughput [34].

The last feature of the first part of the thesis is the introduction of QoS into the campus network. Through simulations, we showed that the FIFO queuing delay in all scenarios is higher than the WFQ ones. The basic reason is that, in the FIFO all packets (from different classes) are enqueued to the same queue, therefore all the differentiated packets has the same queuing delay as the best-effort packets. The WFQ algorithm gives a separate queue and different weight (priority) to each differentiated packet and also to the best-effort packets. Having more queues means that packets are processed faster, and therefore the WFQ algorithm has much less queuing delay for the differentiated packets than the FIFO one. The overall results proved that QoS improves response times and shortens the delay in particularly for the application with the highest priority, which in our study were assigned to DB queries.

Thus, in the first part of the thesis we developed a model that captures the key characteristics of campus networks, including the bursty nature of traffic, the segmentation of our LANs, and the most important bandwidth allocation policies. We then developed a feasible performance methodology for the model that (i) gives good first-cut estimates of end-to-end response times and link utilization, and (ii) requires the use of quality of service for differentiated campus data traffic. In addition to reducing the response times it was shown that the expected waiting times (i.e., queue delay) of our heavy users is reduced by incorporating the suggested control mechanism of QoS, which can accommodate most of the well-known applications traffic usually flowing through a campus network.

In the second part of the thesis, we focus on the same design of campus networks but now with intelligent components to facilitate knowledge processing. We suggest a new approach in routing intelligently network packets. The idea is to divide the application layer into layers that deals with the following: the forwarding packet process, the quality of service integrated into the packet, and finally the type of application the packet belongs to. In addition, we suggest a flexible QoS architecture which combines differentiated traffic via ToS with a known queue mechanism WFQ/RED and a routing mechanism RIP to accomplish efficiently the routing process.

The suggested system, an intelligent educational system (IES), is implemented as an enterprise network convergence of IP and IN technologies that provides advanced knowledge processing services. The generic algorithm for advanced routing is based on

the identification of the traffic according to the application it belongs to. This identification of the packets is the ‘IN-triggering’ mechanism that requires a SCP to provide further routing information to the switches. The generic algorithm worked correctly in all the simulated cases.

It is worth recalling that in the recent past, the term Intelligent Network (IN) became popular to describe a network that could be programmed to do things like forward calls, or enable toll-free numbers. Today we use the term intelligent network to describe a network that can be programmed, and that it can "learn" to provide whatever services the customer wants to. IES is an efficient, intelligent, and feasible knowledge processing system based on existing network technology and telecommunication infrastructures and it functions as a ‘negotiator’ between information seeker and information provider. Its output is maximized by matching the correct amount and quality of available knowledge to the user’s necessity of it.

In conclusion, the goal of this research is to suggests a general fine-tuned architecture of an intelligent network - with its signaling and transport methods, and its intelligence of selecting the right set of knowledge data banks (output) after the correct parsing of incoming information request (input). The simulation results appear to follow the realistic behavior of the network under the applied traffic. This study demonstrates that modeling and simulation can be an effective tool for designing and planning a future network under different scenarios. It provides an enabling technology to proactively serve the demands of higher-education users by providing access to sources of information satisfying those demands at anytime and anywhere in the world.

15.1 Future Research

Despite the enormous potential of knowledge-based educational systems and a phenomenal body of research, remarkably few fully operational knowledge-based educational systems have been developed. Only a very few of the existing knowledge-based educational systems [10] has made the difficult transition from the laboratory to the classroom. Nonetheless, as we move toward an all IP (Internet protocol) core and IMS (Internet Protocol Multimedia Subsystem) architecture, the infrastructure is coming together pointing to a not so distant future when intelligent networks as they are supposed to be 'intelligent' will be a reality. What most studies on knowledge-based educational systems agree is that it requires melding advances in computer science, networks, and software engineering with state-of-the-art artificial intelligence to offer an environment that will have a significant impact on both individualized and collaborative learning. It will constitute a quantum leap forward in integrated educational technology and yet expected to remain affordable in both hardware and bandwidth demands.

Beyond the problems investigated in this thesis the following issues can be topics for further research: refinement of the proposed traffic control mechanism, such as traffic conditioner, buffer management, scheduling, security, charging and accounting, dynamic allocation of bandwidth, and performance studies using different traffic models. Another area of research includes investigating the applicability of an emerging technology called deep-packet inspection which allows switches to gain a better view of the data packet's content, thus data is sent to the application or resource that can best process the user's request or any security risk can be avoided. Also of special interest is the case where

heterogeneous sources are present and/or statistical multiplexing is used. It can be equally important the computationally efficient techniques for determining implied costs when heterogeneous sources are present.

It will be interesting exercise to determine the impact of the proposed control mechanisms on the performance when more nodes can be included in the simulation and/or real data traffic can be imported into the model. Word of caution in simulating larger networks, OPNET requires extremely powerful computers, and if not using them, simulation times can become quite long. A number of techniques have been studied elsewhere for accelerating the simulation. One of the most regarded is concurrent (parallel) simulation which exploits the inherent parallelism in telecommunications networks, and it would be challenging to use it in a very large network.

A daunting task is that of designing a ‘super interpreter’ that supports the use of natural language and it is capable of scanning the information, associating it with previous information(s) by pattern matching and lexically scanning the objects that make up the current information. The inferences and evaluations may accrue in the system to give these systems a knowledge profile of the user to customize the system’s response to information request(s). This technology would promote more self-fulfilling forms of learning. For this reason, theory and research on this subject of intelligent knowledge processing, such as the one promoted by this thesis, provide an extremely important source of ideas and improve the well being of everybody.

IES is one of the multimedia information technologies that complements and enhances our quest for knowledge and it can offers several areas for innovative research such as:

(a) information structure – equipment such as ATM switching systems and fiber-optic networks have been constructed to provide broadband multimedia services,

(b) database technology – the amount of data in the database of multimedia information is increasingly growing, thus, requiring intelligent database warehousing and mining processes,

(c) image recognition – when a person's/object's shape, color, and texture are determined, the information can be available to knowledge services, and

(d) human-machine interface – virtual reality, natural language, speech recognition, and handwritten text recognition will provide human-friendly interfaces.

16. Glossary

10BaseT	The technical name for twisted pair Ethernet
802.2	The ISO Standard for Logical Link Control
802.3	The ISO Standard for Ethernet
802.5	The ISO standard for Token Ring
ABR	Available Bit Rate
ADSL	Asymmetric Digital Subscriber Line
API	Application Program Interface
ARP	Address Resolution Protocol
AS	Autonomous System
ASICs	Application Specific Integrated Circuits
ASCII	American Standard Code for Information Interchange
ATM	Asynchronous Transfer Mode
AUI	Attachment Unit Interface
bps	Bits per second
BGP	Border Gateway Protocol
BITNET	Because It's Time Network
BRI	Basic Rate Interface
CBQ	Class-Based Queuing
CBR	Constant Bit Rate
CCITT	Consultative Committee on International Telephone and Telegraph
CDDI	Copper Distributed Data Interconnection
CGI	Common Gateway Interface

CIDR	Classless Inter-Domain Routing
CN	Campus Network
CO	Connection-Oriented
CoS	Class of Service
CRC	Cyclic Redundancy Check
CSMA/CD	Carrier Sense Multiple Access with Collision Detection
DCE	Data Circuit-Terminating Equipment
DTE	Data Terminal Equipment
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
DS-1, DS-3	Speeds of popular point-to-point digital circuits. DS-1 denotes 1.544, and DS-3 denotes 44.736 Mbps. See T1, T3
DSL	Digital Subscriber Line
DSU/CSU	Data Service Unit/Channel Service Unit
DVMRP	Distance Vector Multicast Routing Protocol
EBCDIC	Extended Binary Coded Decimal Interchange Code
ECN	Explicit Congestion Notification
EIGRP	Extended Interior Gateway Protocol
EGP	Exterior Gateway Protocol
ETE	End-To-End
FCS	Frame Check Sequence
FDDI	Fiber Distributed Data Interconnect
FDM	Frequency Division Multiplexing

FIFO	First-In First-Out
FTP	File Transfer Protocol
FTTC	Fiber To The Curb
Gbps	Giga bits per second
GIF	Graphics Interchange Format
HDSL	High-Rate Digital Subscriber Line
HFC	Hybrid Fiber Coax
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
IANA	Internet Assigned Number Authority
ICMP	Internet Control Message Protocol
ICN	International Campus Network
IETF	Internet Engineering Task Force
IGMP	Internet Group Multicast Protocol
IGP	Internet Gateway Protocol
ICS	Internet Connection-Sharing
IN	Intelligent Network
IP	Internet Protocol
IPng	Internet Protocol – The next Generation
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
IPX	Internet Packet eXchange
ISDN	Integrated Services Digital Network

ISO	International Organization for Standardization
ISP	Internet Service Provider
ITU	International Telecommunications Union
ITU-T	International Telecommunications Union, Telecommunication Standardization Sector
Kbps	Kilo bits per second
LAN	Local Area Network
LANE	LAN Emulation
LLC	Logical Link Control
MBONE	Multicast backBONE
Mbps	Mega bits per second
MIB	Management Information Base
MIME	Multipurpose Internet Mail Extensions
modem	MOdulator/DEModulator
MOSPF	Multicast extensions OSPF
MPLS	Multiprotocol Label Switching
MTU	Maximum Transmission Unit
NAT	Network Address Translation
NCN	National Campus Network
NCP	Network Control Protocol
NFS	Network File System
NGI	Next Generation Internet
NIC	Network Interface Card

NNI	Network Node Interface
NSFNET	National Science Foundation Network
OC	Optical Carrier
OC-3	Speed of an optical fiber cable that operates at 155.520 Mbps
OSPF	Open Shortest Path First Protocol
PCM	Pulse Code Modulation
PHB	Per-Hop Behavior
PHP	Perl Helper Pages
POP	Point of Presence
POTS	Plain Old Telephone Service
PPP	Point-to-Point Protocol
PRI	Primary Rate Interface
PSTN	Public Switched Telephone Network
PVC	Permanent Virtual Circuit
QoS	Quality of Service
RARP	Reverse Address Resolution Protocol
RED	Random Early Detection
RFC	Request For Comments
RIP	Routing Information Protocol
RSVP	Resource Reservation Setup Protocol
SCP	Service Control Point
SLA	Service Level Agreement
SMDS	Switched Multi-megabit Data Service

SMTP	Simple Mail Transfer Protocol
SNAP	SubNetwork Attachment Point
SNMP	Simple Network Management Protocol
SPF	Shortest Path First
STDM	Synchronous Time Division Multiplexing
STP	Spanning Tree Protocol
STS	Synchronous Transport Signal
SVC	Switched Virtual Circuit
T1, T3	Designations of popular point-to-point digital circuits. A T1 circuit operates at 1.544 Mbps, and a T3 circuit operates at 44.736 Mbps. See DS-1, DS-3
TCP	Transmission Control Protocol
TCP/IP	The protocol suite used in the Internet. Although the suite contains many protocols, TCP and IP are two of the most important.
TDM	Time Division Multiplexing
TMN	Telecommunications Management Network
ToS	Type of Service
TTL	Time To Live
UBR	Unspecified Bit Rate
UCAID	University Corporation for Advanced Internet Development
UNI	User-to-Network Interface
UDP	User Datagram Protocol
vBNS	very high Bandwidth Network Service

VC	Virtual Circuit
VPI/VCI	Virtual Path Identifier/Virtual Channel Identifier
VPN	Virtual Private Network
WAN	Wide Area Network
WAP	Wireless Access Protocol
Web	A synonym for World Wide Web
WFQ	Weighted Fair Queuing
WWW	World Wide Web
X.24	ITU packet-switching protocol standard
xDSL	A generic acronym used to refer to any of the local loop technologies such as: ADSL, HDSL, etc.

17. Bibliography

[1] AT&T, "Telecommunication Transmission Engineering – Volume I – Principles", Second/Third Edition

[2] AT&T "Telecommunication Transmission Engineering – Volume II – Facilities", Second/Third Edition

[3] AT&T, "Telecommunication Transmission Engineering – Volume III – Networks and Services", Second/Third Edition

[4] M.C.Chow (1996), "Understanding SONET/SDH: Standards and Applications", Andan Publisher, Holmdel, NJ

[5] Mark A. Miller (1998), "Implementing IPv6", John Wiley & Sons

[6] W. Stallings (1987) "Data and Computer Communications", 2nd ed, Macmillian

[7] S.V.Ahamed, V.B.Lawrence (1998), "Intelligent Broadband Multimedia Networks", Kluwer Academic Publishers

[8] E. A. Lee and D.G. Messerschmitt (1994), "Digital Communication", 2nd ed, Kluwer Academic Publishers

[9] S.V.Ahamed, V.B.Lawrence (1997), "Design and Engineering of Intelligent Communication Systems", Kluwer Academic Publishers

[10] J. Wang, K. Nahrstedt, Y. Zhou (2000), "Design and Implement Differentiated Service Routers in OPNET", OPNETWORK 2000

[11] L. Goleniewski (2003) "Telecommunications Essentials", Addison-Wesley

[12] J.Lester, B.Porter (1994), "Designing multimedia knowledge delivery systems. The strong representation paradigm," Working notes of the AAAI Spring Symposium on Intelligent Multi-Media Multi-Modal Systems, pp 64-72

[13] K.Forbus, P.Feltovich (2001), "Smart Machines in Education", The MIT Press

[14] Mil3 Inc., OPNET Technologies Inc (2001), "OPNET IT Guru User's Manual"

[15] NIST (1998), " ATM Simulator: Operation and Programming Guide"

[16] A.Law, W.Kelton (2000), "Simulation Modeling & Analysis", 3rd ed, McGraw-Hill

[17] D.Gross, C.Harris (1997), "Fundamentals of Queuing Theory", John Wiley & Sons

[18] R. Richter, Walrand J. C. (1989), "Distributed Simulation of Discrete Event Systems", Proceedings of the IEEE, Vol 77

[19] C. Harrell, B. Ghosh, et. Al (2004), "Simulation using Promodel", 2nd ed, McGraw-Hill

[20] M. H. MacDougall (1987), "Simulating Computer Systems – Techniques and Tools", The MIT Press

[21] D. G. Kendall (1953), "Stochastic Processes Occurring in the Theory of Queues and Their Analysis by the Method of Imbedded Markov Chains", Annals of Mathematical Statistics 24, pp. 338-54

[22] J. Hamilton Jr., D. Nash, U. W. Pooch (1997), "Distributed Simulation", CRC Press

[23] A. Leon-Garcia, I. Widjaja (2000), "Communication Networks", McGraw-Hill

[24] T. Q. Andry, K. Haller (1998), "Designing Campus Networks", Macmillan Technical Publishing - Cisco Press

[25] D. E. Comer (2001), "Computer Networks and Internets", 3rd ed., Prentice Hall

[26] R. B. Yate, B. R. Neto (1999), "Modern Information Retrieval", Addison-Wesley

[27] V. Paxson, S. Floyd (1995), "Wide-Area Traffic: The Failure of Poisson Modeling", IEEE/ACM Transactions on Networking, 3(3), pp. 226-244

[28] P. Leys, J. Potemans, B. Van den Broeck (2002), "Use of Raw Packet Generator in OPNET", OPNETWORK 2002, Washington DC, USA

[29] W. Simpson, Ed. (1994), "The Point-to-Point Protocol (PPP)", IETF RFC 1661

[30] B. Kerkhoff, M. Kolon (2002), "Traffic Policing in Academic Environments", Juniper Network, PN# 350031-001

[31] L. L. Peterson, B. S. Davie (2000), "Computer Networks – A Systems Approach", 2nd ed., Morgan Kaufmann Publishers

[32] Berkeley's "Commodity Network Bandwidth: Problems and Recommendations" (2002), http://nac.berkeley.edu/bandwidth/

[33] R. A. Richards (2003), "Intelligent Network Configuration Optimization Toolkit (INCOT)", IEEE Aerospace Conference Proceedings

[34] V. Frost, B. Melamed (1994), "Traffic Modeling for Telecommunications Networks", IEEE Communications Magazine, 32(3):70-81

[35] M. Thelwall et al (2002), “Which Academic Subjects have most online impact?”, Online Information Review 27(5), 333-343

[36] C. Jenkins et.al (1998), “Automatic Classification of Web resources using Java and Dewey Decimal Classification”, Elsevier Science Publishers B. V., Proceedings of the Seventh International Conference on World Wide Web 7

[37] J. Davidson, J. Peters (2000), “Voice over IP Fundamentals”, Cisco Press

[38] Ahamed, S. V. and V. B. Lawrence (1998), “Architecture for a computer system used for processing knowledge”: Patent US05465319

[39] Technical Team (2004), “Overview of Network-Based Application Recognition”, Cisco Press

[40] D. de Capite (2006), “Self-Defending Networks: The Next Generation of Network Security”, Cisco Press

[41] M. Thelwall et al (2002), “A Pilot Study for New Subject Classification Process”, Online Information Review 27(5), 356-372

[42] U. Black (2000), “Emerging Communications Technologies”, 3rd ed, Prentice Hall

[43] Hwang et al (1991), “Query-Based Language Applied to Partially Trained Multi-Layer Perceptrons”

[44] P. Oppenheimer and J. Bardwell (2002), “Troubleshooting Campus Networks”, John Wiley & Sons

[45] S. Aidarous and T. Plevyak (2003), “Managing IP Networks”, John Wiley & Sons

[46] J. Sterbenz and J. Touch (2001), “High-Speed Networking”, John Wiley & Sons

[47] A. Zinin (2002), “Cisco IP Routing”, Addison-Wesley

[48] J. Anderson (2002), “Intelligent Networks: Principles and Applications”, IEE Telecommunications Series 46

[49] J. Thörner (1994), “Intelligent Networks”, Artech House

[50] R. Bates and M. Bates (2007), “Principles of Voice & Data Communications”, McGraw-Hill

[51] J. Davidson (2000), “Voice over IP Fundamentals”, Cisco Press