7-27-2001

Predictive performance modeling and simulation

Garth Crosby
Florida International University

DOI: 10.25148/etd.FI14061546

FLORIDA INTERNATIONAL UNIVERSITY

Miami, Florida

PREDICTIVE PERFORMANCE MODELING AND SIMULATION

A thesis submitted in partial fulfillment of the

requirements for the degree of

MASTER OF SCIENCE

in

ELECTRICAL ENGINEERING

by

Garth Crosby

2001

To: Interim Dean Richard Irey
College of Engineering

This thesis, written by Garth Crosby, and entitled Predictive Performance Modeling and Simulation, having been approved in respect to style and intellectual content, is referred to you for judgment.

We have read this thesis and recommend that it be approved.

Dr. Jean Andrian

Dr. Maria Martinez

Dr. Subbarao Wunnava, Major Professor

Date of Defense: July 27, 2001

The thesis of Garth Crosby is approved.

Interim Dean Richard Irey
College of Engineering

Dean Douglas Wartzok
Division of Graduate Studies

Florida International University, 2001

DEDICATION

I dedicate this thesis to my mother, whose love and support has motivated me to excel and to the Almighty, who is the source of my strength.

ACKNOWLEDGMENTS

I would like to thank the members of my committee for their assistance in the completion of this thesis. Their input was most sincerely appreciated. I give special thanks to Dr. SubbaraoWunnava, who sought to guide me through to the completion of my thesis in spite of trying physical ailments. Thanks also to Dr. Jean Andrian and Dr. Maria Martinez for the indispensable input they made towards the completion of this thesis. I must also thank Mrs. Pat Brammer who was instrumental in the success of not only my thesis, but every endeavor that I was involved in during my master's program at FIU.

ABSTRACT OF THE THESIS

PREDICTIVE PERFORMANCE MODELING AND SIMULATION

by

Garth Crosby

Florida International University, 2001

Miami, Florida

Professor Subbarao Wunnava, Major Professor

The purpose of this thesis was to create and simulate a model of an existing Campus network with a view to predict future performance. This thesis also suggests ways in which such a network can be optimized.

Such simulation and modeling can be referred to as "Predictive Performance Modeling'. In this research a model of Florida International University (University Park campus) High Speed Network was created. Simulation of the model was carried out and an ATM Backbone Analysis was done. The results obtained were compared with corresponding results obtained by network performance monitoring and measurement software tools. A strong correlation between measured and simulation results were observed. A more detailed model was also created for the Engineering and Applied Science (EAS) Local Area Network. Various performance parameters results were collected and analyzed. Based on simulated results, predictions were made in regards to the scalability and optimization of this network in light of expected future requirements.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE PAGE

Chapter 1 Introduction

1.1 Communication Systems

Communication systems are becoming ever more complex as technological advances permit ever faster transmission over links of ever greater capacity. Communication networks are organized hierarchically as Local Area Networks (LANs) for communication, e.g. within a building, Metropolitan Area Networks (MANs), which contain geographically close nodes, e.g. within a city, and Wide Area Networks (WANs) for communication over longer distances up to thousand of miles. There may also be other levels in hierarchy, for example using satellites. Communication between any two nodes in such a system may involve many intermediate nodes and meet contention from diverse traffic sources. The number of nodes in a network may run into thousands: they have diverse characteristics ranging from a local workstation to a gateway between sub-networks, and the links between nodes may also be different types, e.g. Ethernet, radio, fiber optics. The design of efficient communication systems, giving at least a minimum specified level of performance, is therefore of paramount importance [1].

1.2 Performance Modeling

Performance models of computer communication systems have been studied for many years with a view to assisting optimization and guiding the design of new generations. To date, however, they have seen limited use in practice: it has been reasonably effective to tune an optimization or new design using educated guesses and experimentation with existing similar systems- even though this process is often unreliable and requires expensive, exclusive use of the resources. Now, however, with the dramatic increase in complexity associated with current systems, such intuitive understanding of system behavior is much harder to acquire. Hence formal models of performance are necessary for efficient and reliable design and/or optimization. There are two main types of models available for this purpose: simulation and stochastic, the latter also often being referred to as analytic [1].

1.3 Analytical Modeling

Analytical modeling has been the most common technique used over the years. It derives its popularity from the fact that if the initial model of a system is accurate and mathematically solvable, then all questions relating to the system's performance can be answered with relative ease. For small systems or for systems that are well understood, accurate analytical modeling is quite feasible. For large systems, an attempt may be made to divide them into small subsystems. When the behavior of a subsystem is statistically independent of the behavior of other subsystems, the subsystem can be modeled independently [2].

Analytical modeling proves inadequate when a large or complex system cannot be conveniently dissected into statically independent subdivisions. As the complexity of a model increases, solvability of mathematical equations decreases, sometimes drastically. Complex models are often solved using numerical techniques, i.e., using computational methods. The advantage of easy answers to all questions no longer exists. Computational models are also potentially expensive [2].

1.4 Simulation

An alternative to analytical modeling is simulation. It may be possible to simulate each component of a system in detail, resulting in an emulation of the system. Alternatively, simulation of the statistical behavior of a component or group of components may be preferred, resulting in a statistical simulation of the system. The particular choice will depend upon the system application and upon the desired degree of detail. Simulation to mean either emulation or statistical simulation, can be performed in hardware, firmware, or software [2].

1.5 Simulation Technique

Statistical simulation using powerful computers has become a popular technique for studying complex systems. Simulation offers an alternative to the analytical approach when the complexity of a system prohibits exact solutions. Simulation also offers a means to evaluate and compare new systems before they are built.

A simulation model describes events, their effects on system components on other events, and their occurrence, based on the underlying statistical behavior. When driven by a proper statistical input stream, a simulation model imitates the system being simulated and generates data equivalent to measurement in a real system experiment [2].

Simulation of a system undergoes three phases:

- Building of the simulation model
- Design of simulation experiments
- Analysis of data generated in simulation experiments

All three phases are equally important in successful usage of the simulation technique.

1.5.1 Simulation Model

It is hardly surprising that the formulation of simulation is critical; it is not obvious, however, that essential to proper formulation is the choice of the level at which a system is simulated. A detail model will increase costs in the development and in the execution of the simulation experience. However, some of the questions may remain unanswered if the detail is not of sufficient depth. A careful trade-off is often required. The level of detail should be fine enough to answer the questions posed and yet be so macroscopic as to ignore details not essential to the results sought.

1.6 Simulators

A simulator can be defined as a tool that allows an analyst to define and execute a simulation model [2]. A simulator may be written explicitly using such languages as C/C++ or Java.

Fundamental to all simulators are the notions of simulated time and statistically controllable events. Any action that alters the state of a simulated system is an event. A simulator predicts, either deterministically or statistically, the time at which an event is to occur. This information is maintained in a data structure called an event list. This list may often be sorted in increasing time order. A simulator also maintains a simulated clock that holds the 'current' time. A simulator operates as follows (Fig. 1.1). The event list is scanned to find the next event . The time value associated with the event found is the value for the clock. The effects of the events are simulated, causing the system state to change. Statistics are updated. The event just simulated may cause future events. Where necessary, new events and corresponding time are predicted, and the event list is updated. Unless certain predefined conditions are met, the simulation cycle is repeated as long as the event list is not empty [2].

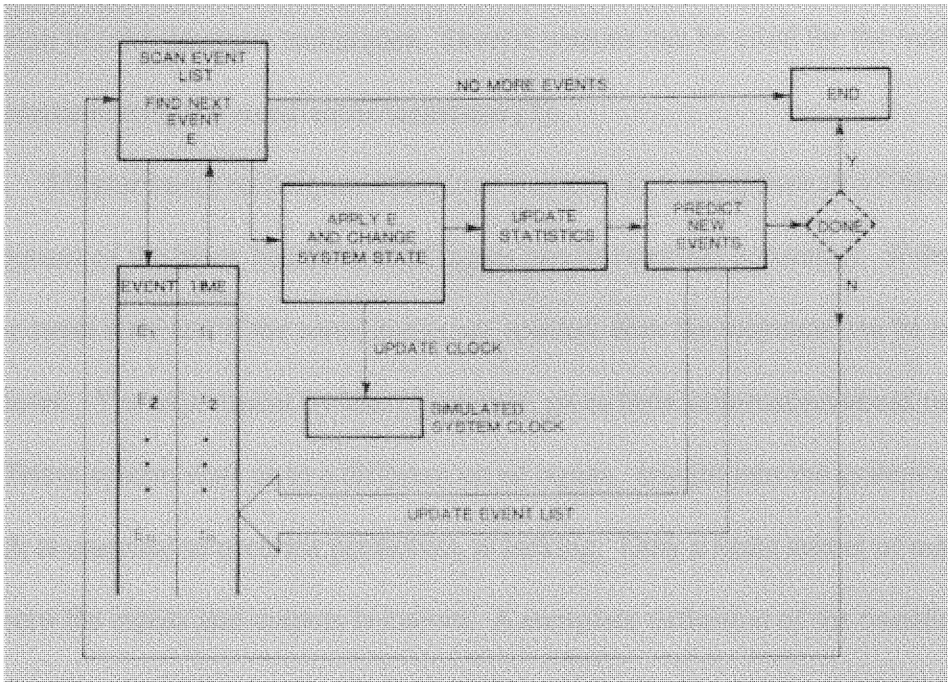


Figure 1.1 Actions of a Simulator

A simulator is designed to imitate some stochastic behavior of the components of a system. If arrival of messages in a system is modeled as a stochastic process, then the time interval between two messages should behave like a random variable drawn from the process. A simulator generates sequences of numbers that follow the rules of the underlying stochastic process. In most simulators, the sequences are not truly random in that they can be exactly duplicated if the same starting value, called seed, is used. Such sequences are known as pseudorandom sequences.

1.7 Measurement Technique

Measurement is perhaps the most obviously appealing technique in that it shows what is and not what should be. When certain inputs are applied to a given system and the outputs measured, they represent the exact mapping from inputs to outputs as performed by the system's transfer function. And yet, measurement serves only a complementary function to other analysis techniques. Several reasons can be cited for the inadequacy of the measurement as a stand-alone technique [2].

Measurement techniques often treat a system or a subsystem as a " black box." Responses of the black box to a set of inputs can be recorded accurately. If the system is in the same state and if the same inputs are applied, then the response can be accurately predicted. The experiment by itself, however, gives little information about responses in a different system state or to a different set of inputs. Insight into the system's working, which is essential to accurate prediction, is lacking. Only a finite subset of mapping between the inputs and outputs is known. A complete definition of the transfer function, and therefore, complete understanding of the system, can be achieved only if the entire input variable space and the entire system state space are exhaustively tested in the measurement experiment. A large input space will reduce practicability of such an experiment. In fact, the experiment may be impossible; because the system is assumed to be a black box, there may be no way to determine the state the system is in.

Measurements serve admirably in conjunction with other methods. In analytical modeling, measurements can be used to validate a model. They can also be used to construct the initial model itself. While measurement techniques are applied to real systems, simulations provide measurements on simulated systems.

The measurement problem is that of determining:

1. what information is needed
2. which measurements will provide the information needed
3. how to obtain the measurements
4. how to interpret the measurements

The first two of these sub-problems are the most difficult ones; in fact, they are a form of a more basic question, namely, "what should constitute the results of analysis? "

Although the discussion refers to measurements, it is equally applicable to the analysis of data gathered from a simulation experiment.

1.8 Measurement Tools

Methods used in measurement can generally be classified into hardware and software tools. They perform functions of sampling, collection, storage, and reporting of statistical data.

1.9 Design of Experiment

An experiment, be it measurement experiment or simulation experiment, consists of many sessions, or runs. The first step in the design of an experiment is the identification of factors, or parameters. These factors can be grouped into two classes, controllable factors and observable factors. An experiment is designed to manipulate the controllable factors in such a fashion that by measuring the observable factors certain inferences can be drawn about the system performance. The feasibility of such inferences and their accuracy is a function of the controllability of the independent factors, interdependence among the observable factors, length of a session, and many such aspects of the experiment.

The length of a session is an important parameter of the experiments. The longer the session, the greater the number of data samples, the smaller the variation in sampled statistics, and the tighter the confidence interval around the estimates [2].

1.10 Motivation for Thesis

Networks and distributed processing systems are of critical and growing importance in business, government, and other organizations. Within a given organization, the trend is toward larger, more complex networks supporting more applications and more users. As these networks grow in scale, two facts become painfully evident:

- The network and its associated resources and distributed applications become indispensable to the organization.
- More things can go wrong, disabling the network or a portion of the network or degrading performance to an unacceptable level [3,4].

Modeling and simulation has become a vital technique for assessing the impact of various design decisions on communication performance. Many companies and government agencies are relying on simulation to help them make intelligent choices while deciding what technologies, equipment, protocols, and algorithms should be employed. The problem is that the different companies use different sets of tools and architectures for evaluating the possible solutions. This means that comparing models and model results is of limited value because each implementation makes its own set of assumptions and has its own implementation of modeled objects and processes. Only if a thorough and detailed understanding of each implementation is known can valid comparison be made. The other problem of having many different implementations is that modules developed by one company are not generally useable in another company's model. In other words, model sharing and reuse are nearly impossible [5].

One approach to solving the problem of multiple model implementations is to define a set of standards for modeling Computer Networks. Optimized Network Engineering Tools (OPNET) is a comprehensive engineering system capable of simulating large communications networks with detailed protocol modeling and performance analysis [6].

In this thesis an existing campus network (Florida International University, University Park Network) is modeled using OPNET. A simulation model was created for the campus wide ATM backbone and traffic analysis was done. This was compared with analysis of measured parameters. Based on the current growth trend of new users and planned migration to VOIP the model was use to predict the corresponding effect on QoS parameters. Thus allowing the facilitation of informed future network upgrade.

In this thesis a detail model for one of the colleges on campus, the Engineering College, was created. A traffic analysis was done. Predictions were also made in regards to necessary bandwidth expansion in order to offer the required QoS in the near future.

Chapter 2 Communication Overview

The 1970s and 1980s saw a merger of the fields of computers science and data communications that profoundly changed the technology, products, and companies of the now combined computer-communication industry.

The computer-communications revolution has produced several remarkable facts:

- There is no fundamental difference between data processing (computers) and data communications (transmission and switching equipment).
- There are no fundamental differences among data, voice, and video communications.
- The distinction among single-processor computer, multiprocessor computer, local network, metropolitan network, and long-haul network has blurred [3].

2.1 A Communication Model

The fundamental purpose of a communication system is the exchange of data between two parties. Figure 2.1. depicts an example, which is communication between a workstation and a server over a public telephone network.

The key elements of the model are as follows:

- Source- This device generates the data to be transmitted.
- Transmitter- This device transform (encode) the data into a form that can be transmitted. Then transmits the data.
- Transmission System- This can be a single line connecting source and destination or a complex network through which the encoded data traverse.

- Receiver- This device accepts the signal from the transmission system and converts it into a form that can be handled by the destination device. For example, a modem will accept an analog signal coming from a transmission system and convert it into a digital bit stream.

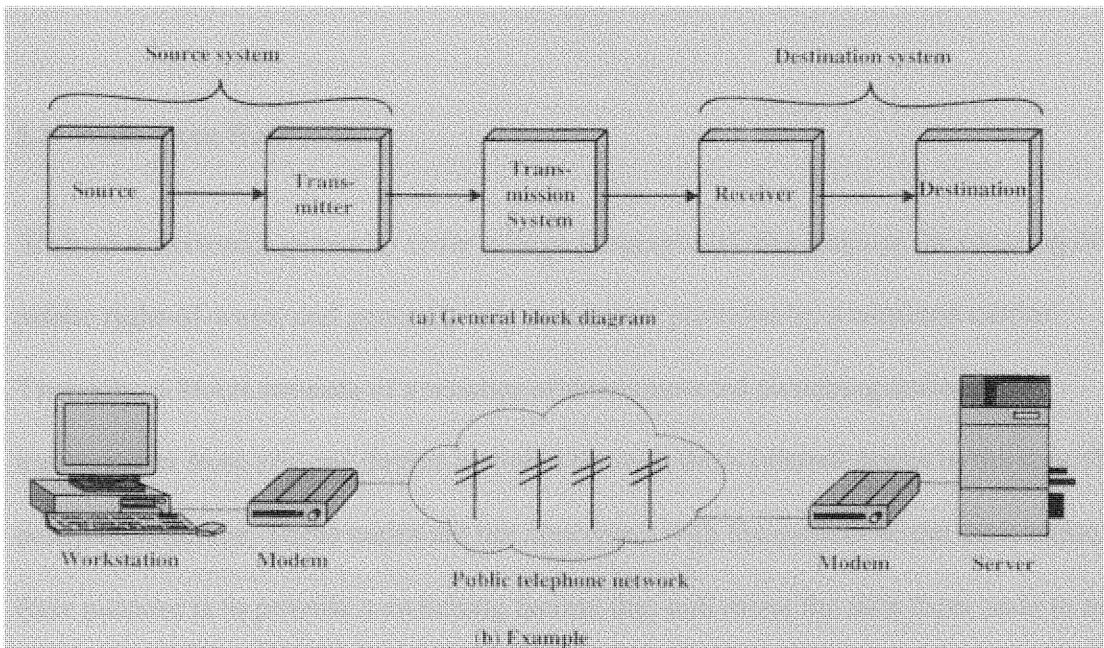


Figure 2.1 Simplified Communications Model

2.2 Data Communication Networking

It is often impractical for two devices to be directly, point-to-point connected. This is so for the following reasons:

- The devices are very far apart. It would be very expensive, for example, to connect a dedicated link between two devices thousands of miles apart.
- There is a set of devices, each of which may require a link to many of the others at various times.

The solution to this problem is to attach each device to a common communication network. Such communication networks are generally classified: wide area networks (WANs) and local area network. The distinction between the two, both in terms of technology and application, has become somewhat blurred in recent years. Nonetheless, the classification provides a useful framework.

2.3 Wide Area Networks

Wide area networks generally cover a large geographical area, require the crossing of public right-of-ways, and generally rely on circuits provided by a common carrier. Typically, a WAN consists of a number of interconnected switching nodes. A transmission from any one device is routed through these internal nodes to the specified destination device. These nodes are not concerned with the content of the data; rather their purpose is to provide a switching facility that will move the data from node to node until they reach their destination.

Traditionally, WANs have been implemented using circuit switching or packet switching technologies. In more recent times frame relay, ATM and SONET have assumed major roles.

2.3.1 Circuit Switching

In a circuit-switching network, a dedicated communication path is established between the source and destination through the nodes of the network. The communication path is formed by a connected sequence of links between the nodes. On each link, a logical channel is dedicated to the connection. Data generated by the source node are transmitted along the dedicated path as fast as possible. At each node, incoming data are routed or switched to the appropriate outgoing channel without delay. The telephone network is a common example of circuit switching network.

2.3.2 Packet Switching

In packet switching it is not necessary to dedicate transmission capacity along a path through the network. Data are sent out in a sequence of small chunks, called packets. Each packet is passed through the Network from node to node along some path leading from source to destination. At each node, the entire packet is received, stored briefly, and then transmitted to the next node. Packet-switching networks are commonly used for computer-to-computer communications.

2.3.3 Frame Relay

Packet switching was developed at a time when digital long-distance transmission facilities exhibited a relatively high error rate compared to today's facilities. As a result, there is a considerable amount of overhead built into packet-switching schemes to compensate for errors. The overhead includes additional bits added to each packet to introduce redundancy and additional processing at the end stations and the intermediate switching nodes to detect and recover from errors.

With modern high-speed telecommunication systems, this overhead is unnecessary and counterproductive. It is unnecessary because the rate of errors has been dramatically lowered and any remaining errors can easily be caught in the end systems by logic that operates above the level of the packet-switching logic. It is counterproductive because the overhead involved soaks up a significant fraction of the high capacity provided by the network.

Frame relay was developed to take advantage of these high data rates and low error rates. Whereas the original packet-switching networks were designed with a data rate to the end user of about 64 kbps, frame relay networks are designed to operate efficiently at user data rates of up to 2 Mbps. The key to achieving these high data rates is to strip out most of the overhead involved with error control [3].

2.3.4 Asynchronous Transfer Mode (ATM)

Asynchronous transfer mode (ATM), sometimes referred to as cell relay, is a culmination of all of the developments in circuit switching and packet switching over the past 25 years.

ATM can be viewed as an evolution from frame relay. The most obvious difference between frame relay and ATM is that frame relay uses variable-length packets, called frames, and ATM uses fixed length packets, called cells. As with frame relay, ATM provides little overhead for error control, depending on the inherent reliability of the transmission system and on higher layers of logic in the end systems to catch and correct errors. By using fixed packet length, the processing overhead is reduced even further for ATM compared to frame relay. The result is that ATM is designed to work in the range of 10s and 100s of Mbps, and in the Gbps range.

ATM can also be viewed as an evolution from circuit switching. With circuit switching, only fixed-data-rate circuits are available to the end system. ATM allows the definition of multiple virtual channels with data rates that are dynamically defined at the time the virtual channel is created. By using small, fixed-size cells, ATM is so efficient that it can offer a constant-data rate channel even though it is using a packet switching technique. Thus, ATM extends circuit switching to allow multiple channels with the data rate on each channel dynamically set on demand.

2.3.5 ISDN and Broadband ISDN

Merging and evolving communications and computing technologies, coupled with increasing demands for efficient and timely collection, processing, and dissemination of information, are leading to the development of integrated systems that transmit and process all types of data. A significant outgrowth of these trends is the integrated services digital network (ISDN).

ISDN, although first published in 1968 by the International Telecommunication Union (ITU), didn't become a standard until 1984 when ITU published the Red Books. Since then new recommendations have been published every four years. The North American version of ISDN is driven by American National Standards Institute (ANSI), and is slightly different from that of ITU [7].

The ISDN is designed to replace existing public telecommunication networks and deliver a wide variety of services. The ISDN is defined by the standardization of user interfaces and implemented as a set of digital switches and paths supporting a broad range of traffic types and providing values-added processing services. In practice, there are multiple networks, implemented within national boundaries, but from the user's point of view, there is a single, uniformly accessible, worldwide network.

Despite the fact that ISDN has yet to achieve the universal deployment hoped for, it is already inn its second generation. The first generation, sometimes referred to as narrowband ISDN, is based on the use of a 64-kbps channel as the basic unit of switching and has a circuit-switching orientation. The major technical contribution of the narrowband ISDN effort has been frame relay. The second generation, referred to as broadband ISDN, supports very high data rates (hundreds of Mbps) and has a packet-switching orientation. The major technical contribution of the broadband ISDN effort has been asynchronous transfer mode (ATM) [3].

2.3.6 SONET

T1 was designed by the Bell Telephone Laboratories back in the 1960s to transmit 24 voice channels digitally over metallic wires. T3 technology, which was an extension of T1, was then introduced to support transmission of 672 voice channels over microwave systems. Since T1 and T3 were both based on the transmission of electrical signals, a new technology that was more appropriate for transmission of optical signals was needed.

This technology was to be designed so that the problems which were inherent in the T-carrier systems would be minimized, and thus make it easier to network.

In 1985 Telcordia provided a solution for these issues in a specification called SONET (Synchronous Optical Network). Since then, SONE has become standardized by ANSI and as SDH (Synchronous Digital Hierarchy) by ITU.

SONET is a multiple-level protocol used to transport high- speed signals using switched synchronous multiplexing. It is the only standard for high-speed fiber systems, which can become the vehicle for making future-generation services such as broadband ISDN, ATM, and others a reality [7].

2.4 Local Area Networks

As with WANs, a LAN is a communications network that interconnects a variety of devices and provides a means for information exchange among those devices. There are several key distinctions between LANs and WANs:

1. The scope of the LAN is small, typically a single building.
2. The entire LAN is normally owned by the same organization. Wan on the other hand typically is not owned by one organization.
3. The internal data rates on LANs are typically much greater than those of WANs.

Traditionally, LANs make use of a broadcast network approach rather than a switching approach. With a broadcast communication network, there are no intermediate switching nodes. At each station, there is a transmitter/receiver that communicates over a medium shared by other stations. A transmission from any one station is broadcast to and received by all other stations. Data are usually transmitted in packets. Because the medium is shared, only one station at a time can transmit a packet.

In recent times switched LANs, in particular Ethernet LANs, have appeared. ATM LANs are also examples of switched LANs.

2.5 Protocols

Basic communication hardware consists of mechanisms that can transfer bits from one point to another. However, using raw hardware to communicate is analogous to programming by entering 1s and 0s – it is cumbersome and inconvenient. To aid programmers, computers attached to a network use complex software that provides a convenient, high-level interface for applications. The software handles most low-level communication details problems automatically, making it possible for application to communicate easily. Thus, most application programs rely on network software to communicate; they do not interact with network hardware directly [8].

All parties involved in a communication must agree on a set of rules to be used when exchanging messages. A Network protocol or computer communication protocol is a set of rules that governs how computers communicate with each other.

2.5.1 The OSI Reference Model

Application Layer
Presentation Layer
Session Layer
Transport Layer
Network Layer
Data Link Layer
Physical Layer

Figure 2.2 The ISO 7 –layer Reference Model

The open systems interconnection (OSI) model was developed by the International Organization for Standardization (ISO) as a model for a computer communications architecture, and as a framework for the development of network protocol standards. It is a seven-layer model as shown in figure 2.2.

The purpose of each layer can be summarized as follows:

- Application Layer– Provides access to the OSI environment. It specifies how one particular application uses a network.
- Presentation Layer- Provides independence to the application processes from differences in data representation by translating data representation on one computer to its representation on another.
- Session Layer- Specifies how to establish, manage and terminate a communication session with a remote computer.

- Transport Layer- Provides reliable, transparent transfer of data between end points; provides end-to-end error recovery and flow control.
- Network Layer – specify how addresses are assigned and how packets are routed through a network.
- Data Link Layer- Provides for the reliable transfer of information across the physical link; sends blocks (frames) with the necessary synchronization, error control, and flow control.
- Physical Layer- Concerned with the transmission of unstructured bit stream over physical medium; deals with the mechanical, electrical, functional, and procedural characteristics to access the physical medium.

Chapter 3 Overview Of FIU High-Speed Network Infrastructure

3.1 Overview

Florida International University's network, FIUnet, is a multi-campus network providing LAN, WAN and Internet services to more than 6000 networked computers on 120 LANs. Port speeds on LAN access devices range from 10Mbps shared ethernet up to 155Mbps on ATM switches. The cabling infrastructure on each campus is a rooted star topology that adheres to the ANSI TIA/EIA standards [1]. As shown in fig 3.1, the FIU Network Backbone, backbone switches in each building are connected to core switches on each campus using optical fiber cable. New buildings, or buildings that have had their network infrastructure upgraded, have ATM switches from IBM and FORE connected to an IBM 8265 switch at the core, using single or dual OC3 trunks. Older buildings with backbones not yet upgraded with ATM, have Cisco Catalyst 3000s in each of their building entrance facilities connected back to a Cisco Catalyst 5000 switch at the core, using full-duplex fast-ethernet trunks [9].

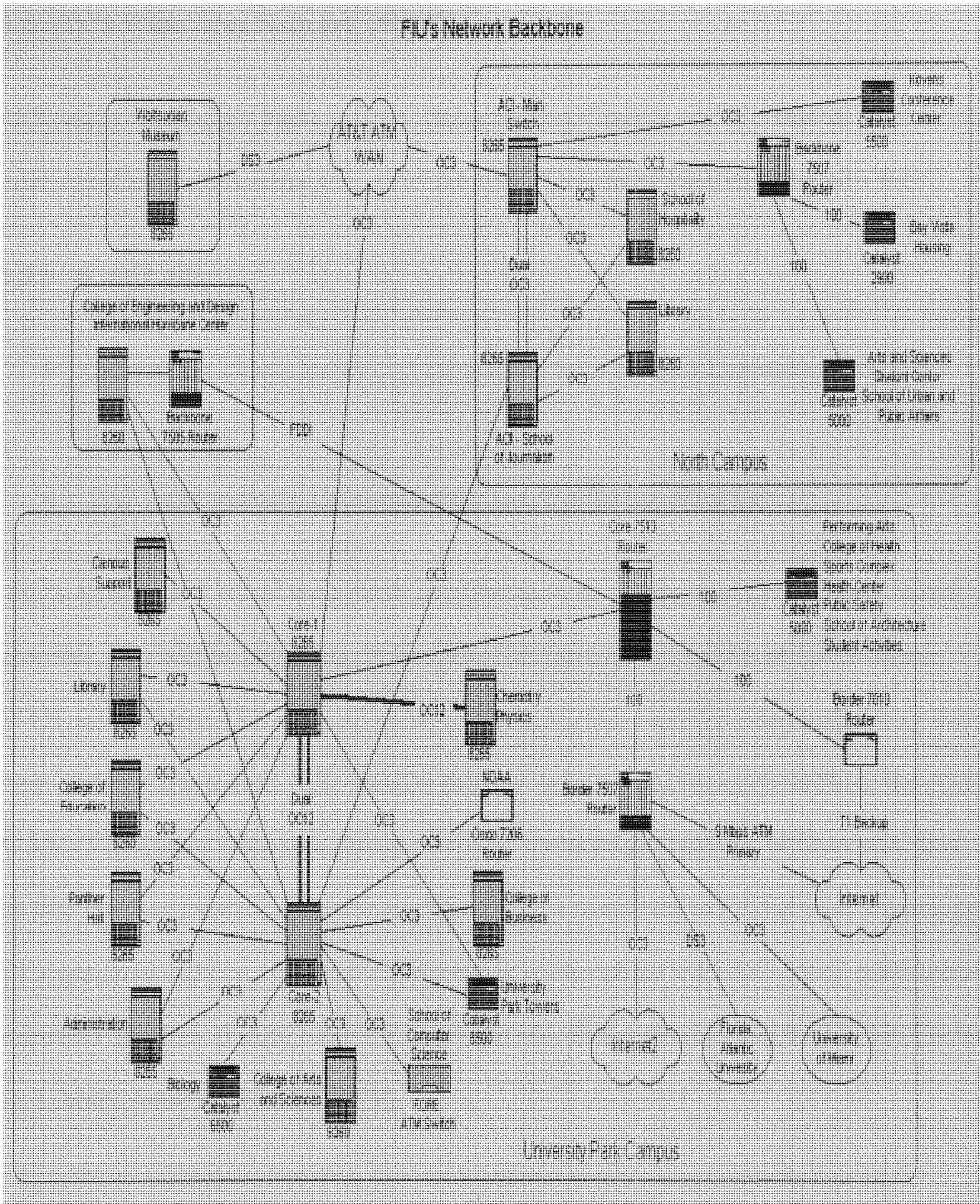


Figure 3.1 FIU's Network Backbone

3.2 FIU's Wide Area Network (WAN)

FIU's main (University Park) campus, Biscayne Bay Campus and Wolfsonian museum is inter-connnected via a wide-area ATM network. University Park and Biscayne Bay

3.2.1 WAN Links

- Campus are inter-connected at OC3 speed links connect University Park and Biscayne Bay
- DS3 links connects University Park and Wolfsonian Museum .

FIU's ATM-based WAN is currently used for its inter-campus data (best-effort) traffic; however, FIU will be migrating its inter-campus video services from an ISDN-based WAN using PictureTEL to its ATM WAN using MPEG-2 quality video technology from First Virtual Corporation. FIU plans to migrate its inter-campus voice traffic to the ATM WAN. FIU's other wide-area services are over frame relay. Institute of Government is an FIU institute located in Downtown Miami that's connected to FIUnet over a fractional T1 frame relay service. FIU maintains peering relationships with the Florida Information Resource Network (FIRN, Florida's statewide education network) and the Florida Center for Library Automation (FCLA) [9].

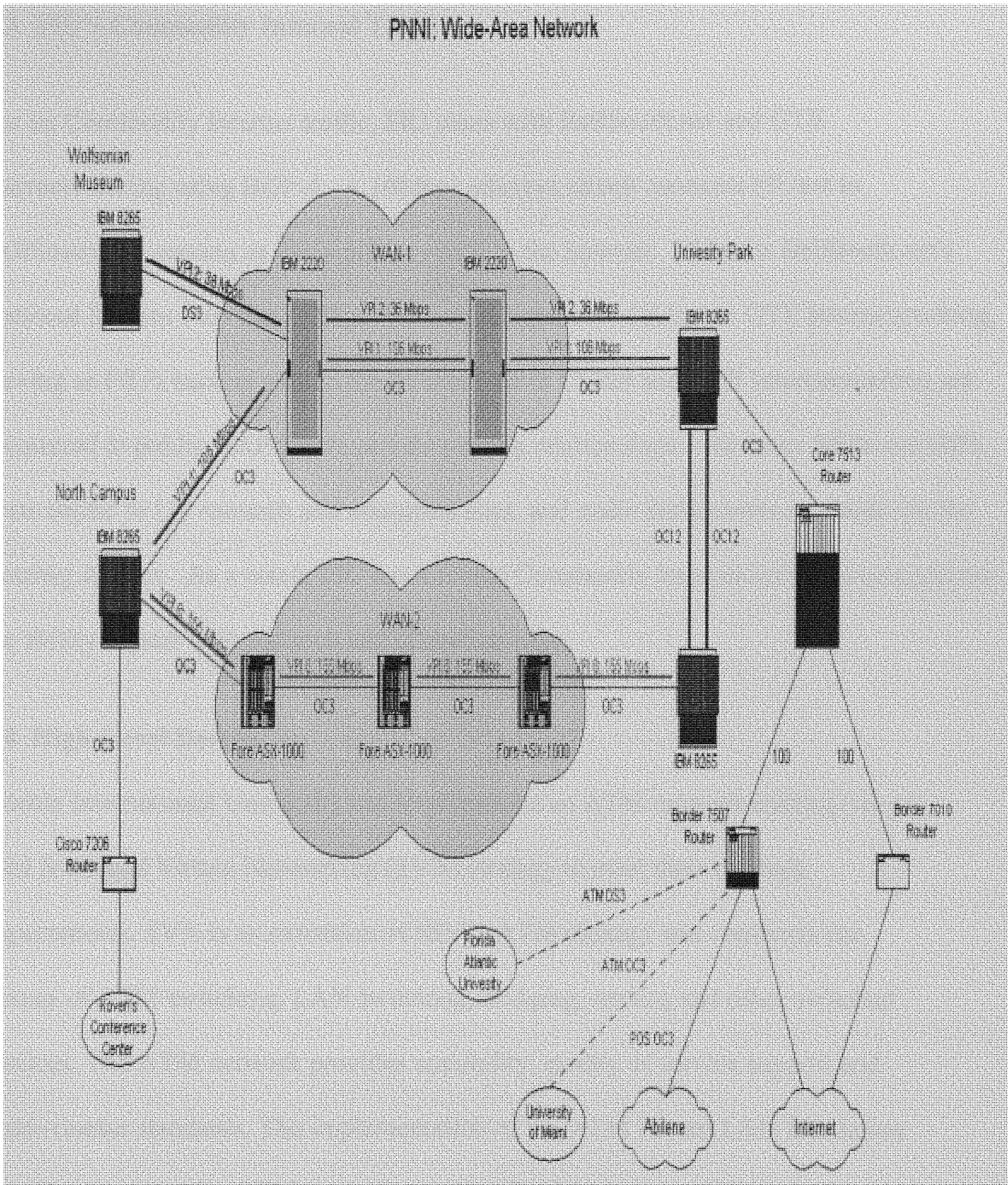


Figure 3.2 PNNI: Wide Area Network

3.3 FIU's Commodity Internet Traffic

For its Commodity Internet traffic, FIU peers with BellSouth.net and GTE Internetworking using BGP-4. A 10Mbps ethernet circuit connects FIUnet to BellSouth.net, serving as FIU's primary path to the Internet. UUnet is BellSouth.net's National Service Provider (NSP). A T1 circuit connects FIUnet to GTE Internetworking, serving as FIU's backup path to the Internet. Both UUnet and GTE Internetworking are connected to public NAPs, have private peering relationships with many NSPs and ISPs, route traffic to US research and education sites, and make these routes available via the Routing Arbiter. As a result, FIU has a robust Commodity Internet infrastructure and has the ability to peer with research and education sites.

3.4 High-Level Design

Florida International University has implemented an ATM network making use of IBM Switched Virtual Networking (SVN) strategy to meet their network requirements. SVN is an extension of IBM's ATM strategy and combines the virtues of ATM switching, LAN switching, bridging, routing and other switched services. The ATM network provides scalability, quality of service support and multi-vendor standards. The implementation of the ATM Forum LAN Emulation specification over the ATM network provides for the interoperability of the network with the legacy network. The ATM campus network at Florida International University is designed to be implemented in phases, co-existing with the legacy network while enabling the implementation of new applications requiring quality of service such as full-motion video and voice [10].

3.5 The Core Network

The design of the ATM network consists of a campus-wide core network that interconnects building backbone networks and the wide area links to remote campuses. The core network is implemented with IBM 8260 ATM switches interconnected with fiber optic links. The long term intention is to interconnect the IBM 8260 core switches in a meshed topology, to provide redundancy in case one of the fiber links is lost. In the initial phases the core network has been interconnected in a star topology using the existing fiber optic links radiating out from the PC building. The bandwidth on the links between the ATM switches can be scaled up in multiples of 155 Mbps or 622 Mbps, by aggregating more fibers into the link between two IBM 8260s, as the growth in traffic volumes require. Connected to the core is a router with an ATM interface and multiple Ethernet interfaces. The purpose of this router is to route data traffic from the legacy network to devices on the ATM network, and vice versa [10].

There are a total of 3,113 ports connected to FIU's ATM network: 1,002 switched ethernet; 530 fast ethernet; 1,284 ATM25; and 297 ATM155 ports.

3.6 Protocols Implemented

All ATM switches run PNNI phase 1 as their signaling protocol. Upgrading to PNNI-1 has allowed FIU to build a standards-based interoperable ATM network that is robust (as a result of PNNI's crankback features) and capable of peering with other peer groups, such as vBNS. Clients and servers on the ATM network use ethernet-based LAN Emulation (LANE) to access services. LANE is provided by IBM's Multi-protocol Switched Server (MSS). MSSs are connected to core and backbone ATM switches and provide a distributed emulated LAN (ELAN) service across the ATM network. ELAN services on the ATM network are very robust because each ELAN component (LECS, LES, BUS and LEC) is redundant with primary and backup components running on multiple MSSs. In the event of a primary ELAN failure, clients automatically join the backup ELAN without loss of service. Classical IP (RFC 1577) is also available on the ATM network and it is used for management of the ATM network devices. OSPF is the IP routing protocol of the ATM and ethernet networks. Variable Length Subnet Masks (VLSM)s are used to efficiently allocate IP address blocks. In addition to IP, AppleTalk is also routed [9]

3.7 Building Backbone

A building backbone network typically consist of a number of workgroup concentrators in the telecommunications closets on the different floors connected to an IBM 8260 ATM switch acting as a cross-connect device. In some buildings this cross-connect device may be the IBM 8260 switch that forms part of the core network. The work group concentrators are connected to the cross-connect ATM switch with fiber optic cables supporting transmission speeds of 155 Mbps. The workgroup concentrators in the telecommunications closets are a combination of IBM 8285 ATM switches supporting 25-Mbps links to workstations with ATM adapters and Ethernet switches with an ATM uplink to the building cross-connect switch and ports supporting category 5 cabling to legacy Ethernet workstations and servers. This mix of work group concentrators in the telecommunications closet enables ATM workstations to be installed and supported where required while connectivity to legacy workstations and servers over the ATM network will be maintained during the migration phases [10].

3.8 FIU's Network Upgrade

The following high-priority initiatives are in the process of being implemented:

- Build a fiber core linking four (4) buildings and forming a ringed topology. Each core building will be an aggregation point for surrounding buildings to connect to one or more core buildings.
- Install additional multimode and single-mode fiber for present and future requirements.
- Install ATM switches at each core building that are fully redundant, scaleable, and capable of supporting multi-service applications with variable Quality of Service (QoS) requirements.

3.9 Modem Pool

The FIU modem pool (305-348-3282) consists of 276 digital US Robotics modems. This modem pool supports connection speeds of up to 56 kbps (with V.90 and X2) and telnet access to local hosts as well as PPP access to the Internet. PPP access today is restricted to FIU faculty and staff and to students whose coursework requires Internet access. In addition to its own modem pool, FIU has a contract with IBM to offer local, national and international dialup access to FIUnet and the Internet. This allows faculty, staff and students to access FIU's network and the Internet from almost anywhere on the planet. FIU is working with BellSouth as a test site for its ADSL deployment in Miami.

3.10 FIU's Technical Support Services

FIU has a functioning Network Operations Center (NOC) whose mission is to provide proactive network monitoring services, and an efficient problem determination and resolution process. The NOC serves as the nerve center for proactive monitoring of FIUnet and its resources, and is the central point of contact for reporting and disseminating information affecting FIUnet services. There are seven (7) dedicated full-time network engineering staff assigned to the support of the University's network. There are 38 other full-time people distributed across the University in colleges and departments responsible for the administration and operation of LANs that connect to the University's backbone [9].

3.11 FIU's Network Management Tools

SNMP-based network management tools are used to monitor the network. These tools provide graphical representations of the network and alarm notification in case there is loss of connectivity to a device; moreover, they automatically page network troubleshooting staff if there's loss of connectivity to a critical resource. Using Multi

Router Traffic Grapher (MRTG) utilization data made viewable using a web browser.

NetFlow is used to further understand backbone traffic [9].

Chapter 4 Asynchronous Transfer Mode

In order to deliver new services such as video conferencing and video on demand, as well as provide more bandwidth for the increasing volume of traditional data, the communications industry introduced a technology that provided a common format for services with different bandwidth requirements. This technology is Asynchronous Transfer Mode (ATM). As ATM developed, it became a crucial step in how companies deliver, manage and maintain their goods and services.

4.1 ATM Technology

Asynchronous Transfer Mode (ATM) is the world's most widely deployed backbone technology. This standards-based transport medium is widely used within the core, at the access and in the edge of telecommunications systems to send data, video and voice at ultra high speeds.

ATM is best known for its easy integration with other technologies and for its sophisticated management features that allow carriers to guarantee quality of service. These features are built into the different layers of ATM, giving the protocol an inherently robust set of controls.

Sometimes referred to as cell relay, ATM uses short, fixed-length packets called cells for transport. Information is divided among these cells, transmitted and then re-assembled at their final destination.

4.2 ATM Cell

The structure of the user –network-interface (UNI) cell header is shown in the following illustration.

4	8 bits	
GFC	VPI	
VPI	VCI	
VCI		
VCI	PT (3 bits)	CLP
HEC		
Information (48 bytes)		

Figure 4.1 User-network-interface (UNI) Cell header

The function of the fields are briefly explained below:

- Generic flow control (GFC)- use to control cell flow at the user-network interface
- Virtual Path Identifier (VPI)- constitutes a routing field for the network
- Virtual Channel Identifier (VCI)- used for routing to and from end users
- Header Error Control (HEC)- used for error control and synchronization
- Payload Type (PT)- indicates the type of information in the information field

Cell Loss Priority (CLP)- used to provide guidance to the network in the event of congestion. A value of 0 indicates a cell of relatively higher priority, which should not be discarded unless no other alternative is available. A value of 1 indicates that this cell is subject to discard within the network.

4.3 BISDN Protocol Reference Model

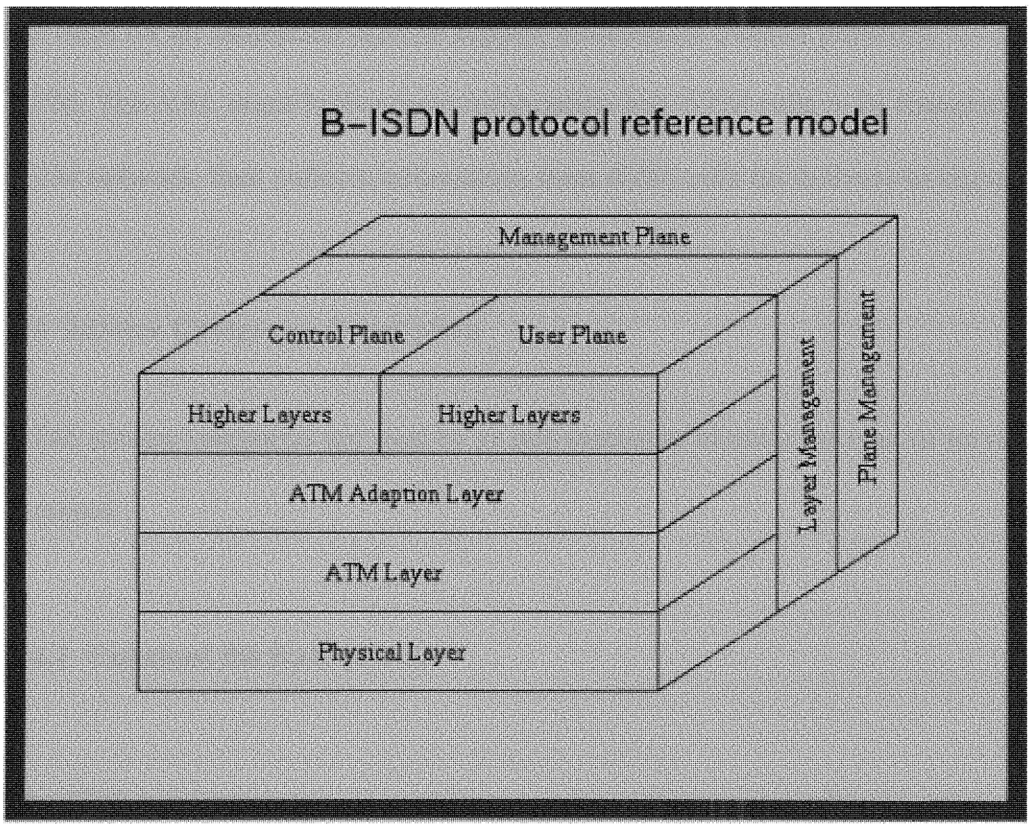


Figure 4.2 B-ISDN Protocol Reference Model

BISDN Protocol Reference Model three planes.

- A User plane for transporting user information.
- A Control plane which is responsible for call control and connection control functions and it contains mainly signaling information.

- A Management plane which contains layer management functions and plane management functions.

Two layers of the protocol architecture relate to ATM functions. There is an ATM layer common to all services that provides packet transfer capabilities, and an ATM adaptation layer (AAL) that is service dependent. The ATM layer defines the transmission of data in fixed-size cells and also defines the use of logical connections. The use of ATM creates the need for an adaptation layer to support information transfer protocols not based on ATM. The AAL maps higher-layer information into ATM cells to be transported over an ATM network, then collects information from ATM cells for delivery to higher layers.

4.4 Service Categories Description

ATM service categories are used by an end system to identify the type of service required.

4.4.1 Constant Bit Rate (CBR)

The CBR service category is used by connections that request a fixed (static) amount of bandwidth, characterized by a Peak Cell Rate (PCR) value that is continuously available during the connection lifetime. The source may emit cells at or below the PCR at any time, and for any duration (or may be silent).

This category is intended for real-time applications, i.e., those requiring tightly constrained Cell Transfer Delay (CTD) and Cell Delay Variation (CDV), but is not restricted to these applications. It would be appropriate for voice and video applications, as well as for Circuit Emulation Services (CES).

The basic commitment made by the network is that once the connection is established, the negotiated QoS is assured to all cells conforming to the relevant conformance tests. It is assumed that cells which are delayed beyond the value specified by Cell Transfer Delay (CTD) may be of significantly less value to the application [11].

4.4.2 Real-Time Variable Bit Rate (rt-VBR)

The real-time VBR service category is intended for time-sensitive applications, (i.e., those requiring tightly constrained delay and delay variation), as would be appropriate for voice and video applications. Sources are expected to transmit at a rate that varies with time. Equivalently, the source can be described as "bursty".

Traffic parameters are Peak Cell Rate (PCR), Sustainable Cell Rate (SCR) and Maximum Burst Size (MBS). Cells which are delayed beyond the value specified by CTD are assumed to be of significantly less value to the application. Real-time VBR service may support statistical multiplexing of real-time sources.

4.4.3 Non-Real-Time (nrt-VBR)

The non-real time VBR service category is intended for applications which have bursty traffic characteristics and do not have tight constraints on delay and delay variation. As for rt-VBR, traffic parameters are PCR, SCR and MBS. For those cells which are transferred within the traffic contract, the application expects a low Cell Loss Ratio (CLR). For all cells, it expects a bound on the Cell Transfer Delay (CTD). Non-real time VBR service may support statistical multiplexing of connections [11].

4.4.4 Available Bit Rate (ABR)

The Available Bit Rate (ABR) is a service category intended for sources having the ability to reduce or increase their information rate if the network requires them to do so. This allows them to exploit the changes in the ATM layer transfer characteristics (i.e., bandwidth availability) subsequent to connection establishment.

It is recognized that there are many applications having vague requirements for throughput: they can be expressed as ranges of acceptable values, e.g., a maximum and a minimum, rather than as an average value (that is typical for the VBR category). To meet this requirement on the establishment of an ABR connection, the end-system shall specify a maximum required bandwidth and a minimum usable bandwidth. These are designated as the Peak Cell Rate (PCR) and the Minimum Cell Rate (MCR), respectively. The MCR may be specified as zero. The bandwidth made available from the network may vary, as it is the sum of an MCR, and a variable cell rate which results from sharing the available capacity among all the active ABR connections via a defined and fair policy. A flow control mechanism is specified which supports several types of feedback to control the source rate. In particular a closed-loop feedback control protocol using Resource Management (RM) cells has been specified in a rate-based framework.

Although no specific QoS parameter is negotiated with the ABR, it is expected that an end-system that adapts its traffic in accordance with the feedback will experience a low Cell Loss Ratio (CLR) and obtain a fair share of the available bandwidth according to a network specific allocation policy. Cell Delay Variation (CDV) is not controlled in this service, although admitted cells are not delayed unnecessarily. ABR service is not (as specified at present) intended to support real-time applications.

A source, destination and network switch behavior is specified by The ATM Forum along with details of the rate-based flow control mechanism.

4.4.5 Unspecified Bit Rate (UBR)

The Unspecified Bit Rate (UBR) service category is a "best effort" service intended for non-critical applications, which do not require tightly constrained delay and delay variation, nor a specified quality of service. UBR sources are expected to transmit non-continuous bursts of cells. UBR service supports a high degree of statistical multiplexing among sources.

UBR service does not specify traffic related service guarantees. Specifically, UBR does not include the notion of a per-connection negotiated bandwidth. There may not be any numerical commitments made as to the cell loss ratio experienced by a UBR connection, or as to the cell transfer delay experienced by cells on the connection

4.5 Role of ATM in the Telecommunication Infrastructure

A telecommunications network is designed in a series of layers. A typical configuration may have utilize a mix of time division multiplexing, Frame Relay, ATM and/or IP. Within a network, carriers often extend the characteristic strengths of ATM by blending it with other technologies, such as ATM over SONET/SDH or digital subscriber line (DSL) over ATM. By doing so, they extend the management features of ATM to other platforms in a very cost-effective manner.

ATM itself consists of a series of layers. The first layer - known as the adaptation layer - holds the bulk of the transmission. This 48-byte payload divides the data into different types. The ATM layer contains five bytes of additional information, referred to as overhead. This section directs the transmission. Lastly, the physical layer attaches the electrical elements and network interfaces.

4.6 ATM as Backbone for Other Networks

The vast majority (roughly 80 percent) of the world's carriers use ATM in the core of their networks. ATM has been widely adopted because of its unmatched flexibility in supporting the broadest array of technologies, including DSL, IP Ethernet, Frame Relay, SONET/SDH and wireless platforms. It also acts a unique bridge between legacy equipment and the new generation of operating systems and platforms. ATM freely and easily communicates with both, allowing carriers to maximize their infrastructure investment.

4.7 ATM in the LAN (Local Area Network)

The LAN environment of a campus or building provides seamless integration to WAN.

4.8 ATM in the WAN (Wide Area Network)

A blend of ATM, IP and Ethernet options abound in the wide area network. But no other technology can replicate ATM's mix of universal support and enviable management features. Carriers inevitably turn to ATM when they need high-speed transport in the core coupled with the security of a guaranteed level of quality of service. When those same carriers expand to the WAN, the vast majority does so with an ATM layer.

Distance can be a problem for some high-speed platforms. Not so with ATM. The integrity of the transport signal is maintained even when different kinds of traffic are traversing the same network. And because of its ability to scale up to OC-48, different services can be offered at varying speeds and at a range of performance levels.

4.9 ATM in the MAN (Metropolitan Area Network)

The MAN is one of the hottest growing areas in data and telecommunications. Traffic may not travel more than a few miles within a MAN, but it's generally doing so over leading edge technologies and at faster-than-lightening speeds.

The typical MAN configuration is a point of convergence for many different types of traffic that are generated by many different sources. The beauty of ATM in the MAN is that it easily accommodates these divergent transmissions, often times bridging legacy equipment with ultra high-speed networks. Today, ATM scales from T-1 to OC-48 at speeds that average 2.5 Gb/s in operation, 10 Gb/s in limited use and spanning up to 40 Gb/s in trials.

4.10 ATM and IP Interworking

ATM and IP are complementary platforms. Discussion often centers around which of the two are more beneficial, but the best-case scenario is actually built on their unified strengths.

Through mass adoption of the Internet, IP has dramatically changed the world with an open, easy-to-use interface to desktop applications. Most carriers maintain an ATM layer over which IP traffic rides because of the traffic-management features that are inherent in ATM. Using ATM, carriers can prioritize traffic to meet end-to-end quality-of-service guarantees. And ATM's broad support of numerous applications is a perfect fit with the multimedia traffic that flows over IP.

4.11 Advantages of Using ATM and IP

By combining IP and ATM, carriers enjoy the reach of IP and management security of ATM. This approach is perhaps the most cost effective way of traffic engineering that meets the stringent standards of today's end users and network carriers.

4.12 Future of ATM and IP

The ATM Forum recognized early on the merits of IP and developed multi-protocol over ATM (MPOA) to address this and other packet technologies. The Forum is also working closely with developers of multi-protocol layer switching (MPLS) in their efforts to apply management techniques to IP traffic. Expect ATM and IP to continue to be the blending of choice that ensures the broadest reach with the highest quality.

4.13 ATM Signaling

The Signaling capability for ATM Networks has to satisfy the following functions:

1. Set up, maintain and release ATM virtual channel connections for information transfer.
2. Negotiate the traffic characteristics of a connection (CAC algorithms are considered for these functions.)

Signaling functions may also support multi-connection calls and multi-party calls. Multi-connection call requires the establishment of several connections to set up a composite call comprising various types of traffic like voice, video, image and data. It will also have the capability of not only removing one or more connections from the call but also adding new connections to the existing ones. Thus the network has to correlate the connections of a call.

A multi-party call contains several connections between more than two end-users like conferencing calls.

4.14 ATM Switching

ATM Switching is also known as fast packet switching. ATM switching node transports cells from the incoming links to outgoing links using the routing information contained in the cell header and information stored at each switching node using connection set-up procedure. Two functions at each switching node are performed by a connection set up procedure.

A unique connection identifier at the incoming link and the link identifier and a unique connection identifier at the outgoing link are defined for each connection.

Routing tables at each switching nodes are set up to provide an association between the incoming and outgoing links for each connection. VPI and VCI are the two connection identifiers used in ATM cells.

4.15 ATM Performance Issues

There are 5 parameters that characterize the performance of ATM switching systems:

1) Throughput

2) Connection Blocking Probability

3) Cell Loss Probability

4) Switching Delay

5) Jitter on the delay

4.15.1 Throughput

This can be defined as the rate at which the cells depart the switch measured in the number of cell departures per unit time. It mainly depends on the technology and dimensioning of the ATM switch. By choosing a proper topology of the switch, the throughput can be increased.

4.15.2 Connection Blocking Probability

Since ATM is connection oriented, there will be a logical connection between the logical inlet and outlet during the connection set up phase. Now the connection blocking probability is defined as the probability that there are not enough resources between inlet and outlet of the switch to assure the quality of all existing as well as new connection.

4.15.3 Cell Loss Probability

In ATM switches, when more cells than a queue in the switch can handle will compete for this queue, cells will be lost. This cell loss probability has to be kept within limits to ensure high reliability of the switch. In Internally Non-Blocking switches, cells can only be lost at their inlets/outlets. There is also possibility that ATM cells may be internally misrouted and they reach erroneously on another logical channel. This is called Cell Insertion Probability.

4.15.4 Switching Delay

This is the time to switch an ATM cell through the switch. The typical values of switching delay range between 10 and 1000 microseconds. This delay has two parts. Fixed Switching Delay and it is because of internal cell transfer through the hardware. Queuing delay and this is because of the cells queued up in the buffer of the switch to avoid the cell loss.

4.15.5 Jitter on the Delay

Also called Cell Delay Variation (CDV) is the probability that the delay of the switch will exceed a certain value.

Chapter 5 Simulation and Measurement Tools

5.1 OPNET

OPNET is a vast software package with an extensive set of features designed to support general network modeling and to provide specific support for particular types of network simulation projects. This section provides only a brief enumeration of some of the most important capabilities of OPNET. Subsequent sections of this chapter provide more detailed information on these features, as well as other aspects of OPNET [6].

5.2 Key System Features

- **Object orientation**. Systems specified in OPNET consist of objects, each with configurable sets of attributes. Objects belong to "classes" which provide them with their characteristics in terms of behavior and capability. Definition of new classes are supported in order to address as wide a scope of systems as possible. Classes can also be derived from other classes, or "specialized" in order to provide more specific support for particular applications.
- **Specialized in communication networks and information systems**. OPNET provides many constructs relating to communications and information processing, providing high leverage for modeling of networks and distributed systems.
- **Hierarchical models**. OPNET models are hierarchical, naturally paralleling the structure of actual communication networks.

• **Graphical specification**. Wherever possible, models are entered via graphical editors. These editors provide an intuitive mapping from the modeled system to the OPNET model specification.

• **Flexibility to develop detailed custom models**. OPNET provides a flexible, high-level programming language with extensive support for communications and distributed systems. This environment allows realistic modeling of all communications protocols, algorithms, and transmission technologies.

• **Automatic generation of simulations**. Model specifications are automatically compiled into executable, efficient, discrete-event simulations implemented in the C programming language. Advanced simulation construction and configuration techniques minimize compilation requirements.

• **Application-specific statistics.** OPNET provides numerous built-in performance statistics that can be automatically collected during simulations. In addition, modelers can augment this set with new application-specific statistics that are computed by user-defined processes.

• **Integrated post-simulation analysis tools**. Performance evaluation, and trade-off analysis require large volumes of simulation results to be interpreted. OPNET includes a sophisticated tool for graphical presentation and processing of simulation output.

• **Interactive analysis**. All OPNET simulations automatically incorporate support for analysis via a sophisticated interactive "debugger".

• **Animation**. Simulation runs can be configured to automatically generate animations of the modeled system at various levels of detail and can include animation of statistics as they change over time. Extensive support for developing customized animations is also provided.

• **Application program interface (API)**. As an alternative to graphical specification, OPNET models and data files may be specified via a programmatic interface. This is useful for automatic generation of models or to allow OPNET to be tightly integrated with other tools [6].

5.3 Typical Applications of OPNET

As a result of the capabilities that were described in the previous sections, OPNET can be used as a platform to develop models of a wide range of systems. Some examples of possible applications are listed below with specific mention of supporting features:

• **Standards-based LAN and WAN performance modeling**. Detailed library models provide major local-area and wide-area network protocols. Configurable application models are also provided by the library, or new ones can be created.

• **Internetwork planning.** Hierarchical topology definitions allow arbitrarily deep nesting of subnetworks and nodes and large networks are efficiently modeled; scalable, stochastic, and/or deterministic models can be used to generate network traffic.

• **Research and development in communications architectures and protocols**. OPNET allows specification of fully general logic and provides extensive support for communications-related applications. Finite state machines provide a natural representation for protocols.

• **Distributed sensor and control networks, "on-board" systems.** OPNET allows development of sophisticated, adaptive, application-level models, as well as underlying communications protocols and links. Customized performance metrics can be computed and recorded, scripted and/or stochastic inputs can be used to drive the simulation model, and processes can dynamically monitor the state of objects in the system via formal interfaces provided by statistic wires.

5.4 OPNET Architecture

OPNET provides a comprehensive development environment for modeling and performance-evaluation of communication networks and distributed systems. The package consists of a number of tools, each one focusing on particular aspects of the modeling task. These tools fall into three major categories that correspond to the three phases of modeling and simulation projects: Specification, Data Collection and Simulation, and Analysis. These phases generally form a cycle, with a return to Specification following Analysis. Specification is actually divided into two parts: initial specification and re-specification, with only the latter belonging to the cycle, as illustrated in the following figure.

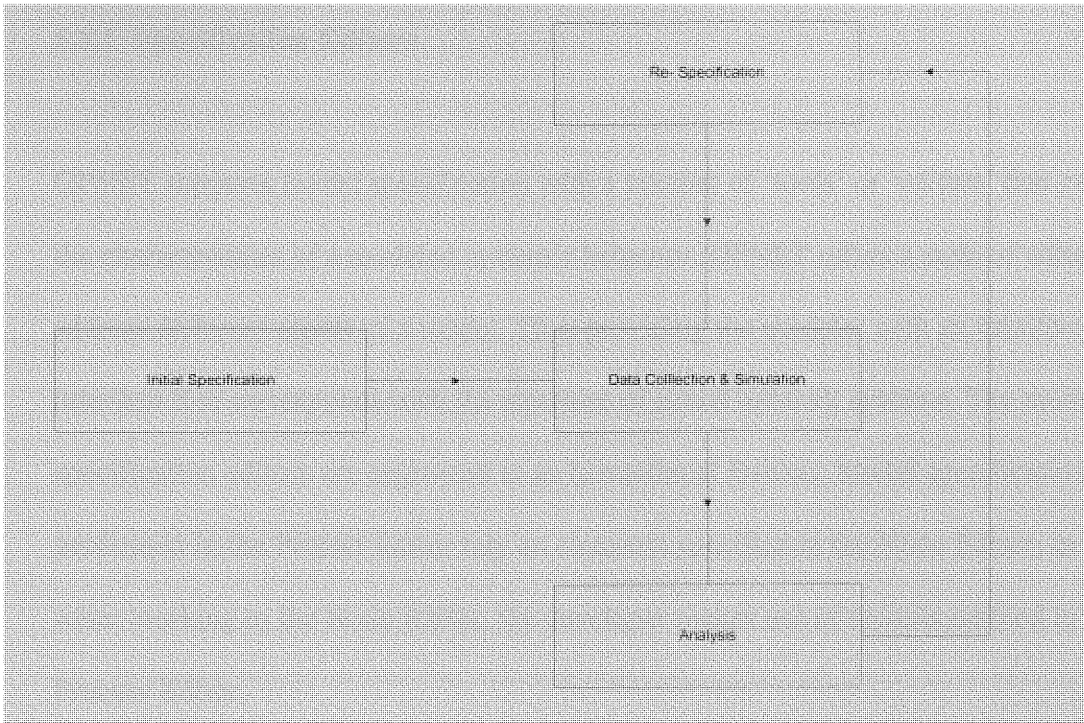


Figure 5.1 Simulation Project Cycle

5.5 Model Specification

Model specification is the task of developing a representation of the system that is to be studied. OPNET supports the concept of model reuse so that most models are based on lower level models developed beforehand and stored in model libraries. Ultimately, however all models are based on the basic concepts and primitive building blocks supplied by the OPNET environment.

5.5.1 Specification Editors

OPNET supports model specification with a number of 'tools' or 'editors' that capture the characteristics of a modeled system's behavior. OPNET offers a suite of editors that address different aspect of a model. Thus providing the modeler with an intuitive interface that corresponds with the structure of actual network systems. The model-specification editors are organized in an hierarchical fashion. Model specifications performed in the Project Editor rely on elements specified in the Node Editor; in turn, when working in the Node Editor, the developer makes use of models defined in the Process Editor.

The other editors can be used to define various data models, and typically tables of values, that are later referenced by process or node level models. The function of the main editors are summarized below:

• **Project Editor -** Develop network models. Network models are made up of subnets and node models. This editor also includes basic simulation and analysis capabilities.

• **Node Editor-** Develop node models. Node models are objects in a network model. Node models are made up of modules with process models.

• **Process Editor**—Develop process models. Process models control module behavior.

• **Link Model Editor**—Create, edit, and view link models.

• **Packet Format Editor**—Develop packet formats models. Packet formats dictate the structure and order of information stored in a packet.

• **ICI Editor**—Create, edit, and view interface control information (ICI) formats. ICIs are used to communicate control information between processes.

• **PDF Editor**—Create, edit, and view probability density functions (PDFs). PDFs can be used to control certain events, such as the frequency of packet generation in a source module.

• **Probe** – For specifying what simulation output data should be collected.

• **Simulation Tool** – For executing simulations.

• **Analysis Tool** – For analyzing simulation results.

5.6 Modeling Domains

The Network, Node, and Process modeling environments are sometimes referred to as the *modeling domains* of OPNET. These domains cover all the hierarchical levels of a model. The remaining specification editors correspond to no particular modeling domain since they mainly support the three principal editors. As mentioned earlier, the capabilities offered by the three modeling domains mirror the types of structures found in an actual computer communication network system; the particular issues addressed by each domain are summarized below in table 5.1 and then briefly described.

Table 5.1 OPNET Modeling Domains

Domain	Editor	Modeling Focus
Network	Project	Network topology described in terms of sub-networks, nodes, links, and geographical context
Node	Node	Node internal architecture described in terms of functional elements and data flow between them
Process	Process	Behavior of processes (protocols, algorithms, applications), specified using finite state machines and extended high level language.

5.6.1 Network Domain

The Network Domain serves to facilitate the definition of the topology of a communication network. The communicating entities are called nodes and the specific capabilities of each node are defined by designating their model. Node models are developed using the Node Editor. Within a single network model, there may be many nodes that are based on the same node model; the term 'node instance' is used to refer to an individual node, in order to distinguish it from the class of nodes sharing the same model.

A network model may make use of any number of node models. OPNET does not place restrictions on the types of nodes that can be deployed in a communication network. In OPNET Modelers are given the flexibility of whereby they can develop their own "library" of node models to use as building blocks for network models. In addition, there are no limits on the number of node models or node instances that a network model may contain.

The Project Editor can provide a geographic context for network model Development (Figure 5.2). You can choose locations on world or country maps for the elements of wide-area networks and can use dimensioned areas for local-area networks. In addition to providing an intuitive environment for deploying the components of a network model, this feature provides an intrinsic notion of distance, which allows automatic calculation of communication delays between nodes.

The basic object used to build network models is the fixed communication node. In OPNET, this is the only type of node available. Fixed nodes can be assigned arbitrary locations, but during a simulation their location may not change. Most nodes require the ability to communicate with some or all other nodes to perform their function in a network model. Several different types of communication link architectures are provided to interconnect nodes that communicate with each other. OPNET provides simplex (unidirectional) and duplex (bi-directional) point-to-point links to connect nodes in pairs and a bus link to allow broadcast communication for arbitrarily large sets of fixed nodes [6].

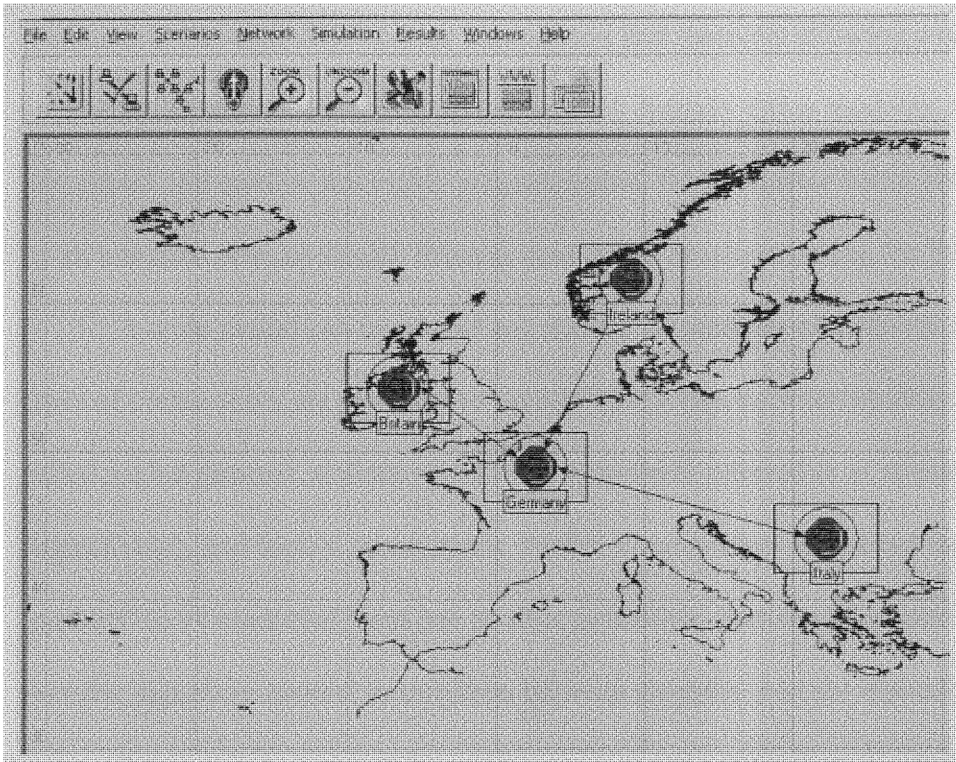


Figure 5.2 Geographical Location of a Network- Network Model

5.6.2 Node Domain

The Node Domain facilitates the modeling of communication devices that can be deployed and interconnected at the network level. In OPNET terms, these devices are called *node*s, and in the real world they may correspond to various types of computing and communicating equipment such as routers, bridges, workstations, terminals, mainframe computers, file servers, fast packet switches, satellites, and so on.

Node models are developed in the Node Editor and are expressed in terms of smaller building blocks called modules. Some modules offer capability that is substantially predefined and can only be configured through a set of built-in parameters. These include various transmitters and receivers allowing a node to be attached to communication links in the network domain. Other modules, called processors and queues, are highly programmable, their behavior being prescribed by an assigned process model. Process models are developed using the Process Editor.

A node model can consist of any number of modules of different types. Three types of connections are provided to support interaction between modules. These are called packet streams, statistic wires (also sometimes referred to as streams and statwires, respectively), and logical associations. Packet streams allow formatted messages called 'packets' to be conveyed from one module to another. Statistic wires convey simple numeric signals or control information between modules, and are typically used when one module needs to monitor the performance or state of another. Figure 5.3 illustrates a typical node model.

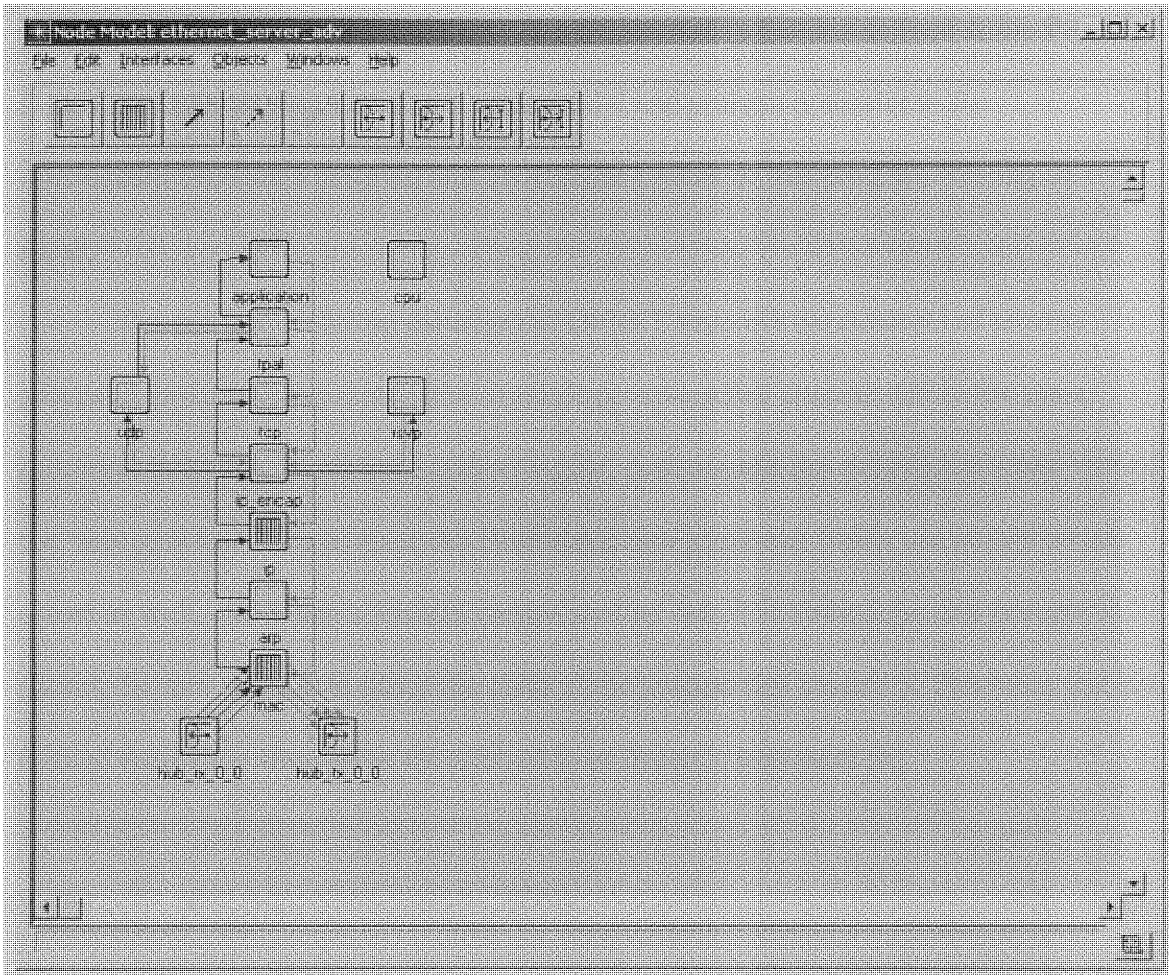


Figure 5.3 Node Model

5.6.3 Process Domain

As mentioned earlier in the discussion of the Node Domain, queue and processor modules are user-programmable elements that are key elements of communication nodes. The tasks that these modules execute are called processes. A process can in many ways be thought of as similar to an executing software program, since it includes a set of instructions and maintains state memory.

Processes in OPNET are based on process models that are defined in the Process Editor. The relationship between process model and process is similar to the relationship between a program and a particular session of that program running as a task.

The process modeling paradigm of OPNET supports the concepts of process groups. A process group consists of multiple processes that execute within the same processor or queue. When a simulation begins, each module has only one process, termed the 'root process'. This process can later create new processes which can in turn create others as well, etc. When a process creates another one, it is termed the new process's parent; the new process is called the child of the process that created it. Processes that are created during the simulation are referred to as dynamic processes [6].

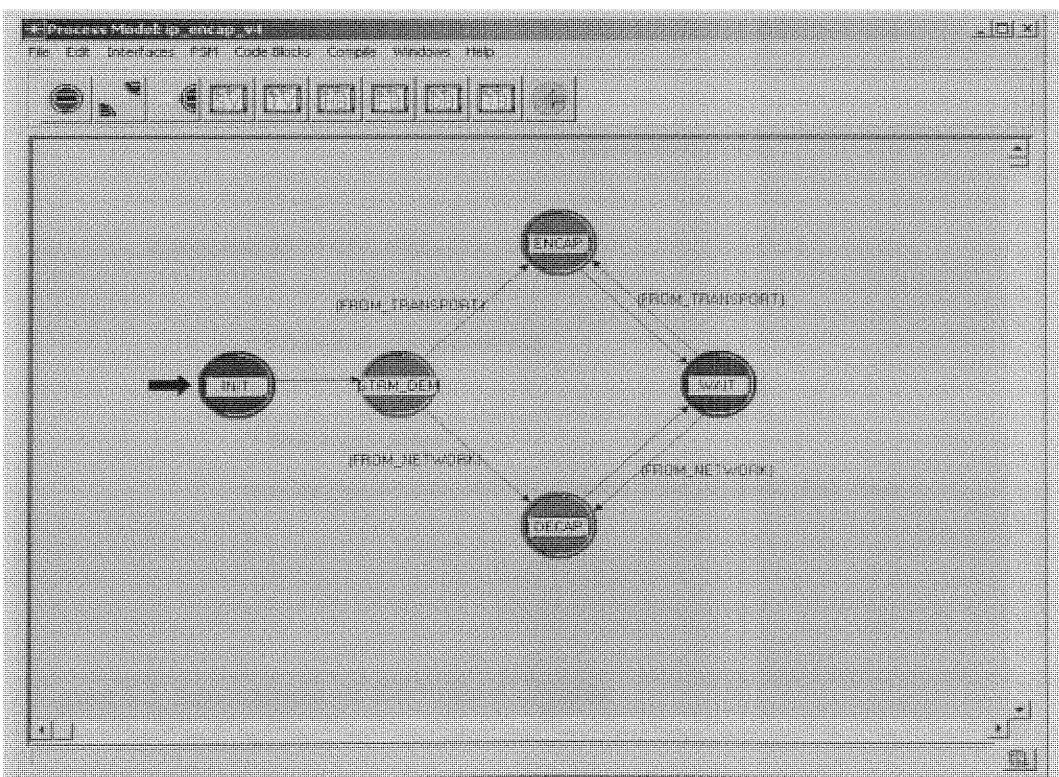


Figure 5.4 An Example of a Process Model

OPNET's Process Editor expresses process models in a language called Proto-C, which is specifically designed to support development of protocols and algorithms. Proto-C is based on a combination of state transition diagrams (STDs), a library of high-level commands known as Kernel Procedures, and the general facilities of the C or C++ programming language. A process model's STD defines a set of primary modes or *states* that the process can enter and, for each state, the conditions that would cause the process to move to another state. The condition needed for a particular change in state to occur and the associated destination state is called a transition.

```
f (intrpt_type == OPC_INTRPT_STRM)
        {
        /** Path Tear message arrived at the node.    **/

        /* Search for a matching Path state with matching session, sender template, incoming interface     */
        path_state_ptr = rsvp_pstate_match_sender_intf_find (path_state_list_ptr, packet_fields_ptr->session_ptr,

                                sender_ptr, in_interface, &list_index);

        if (path_state_ptr == OPC_NIL)
                {
                /** There is no matching path state.          **/

                /* Write a log message.                        */
                rsvp_log_msg_not_path_state_found (packet_fields_ptr, RsvpC_Path_Tear);
                }
        }
```

Figure 5.5 An example of the Proto-C Syntax

5.7 Data Collection and Simulation

The objective of most modeling efforts is to obtain measures of a system's performance or to make observations concerning a system's behavior. OPNET supports these activities by creating an executable model of the system. Provided that the model is sufficiently representative of the actual system, OPNET allows realistic estimates of performance and behavior to be obtained by executing simulations. Several mechanisms are provided to collect the desired data from one or more simulations of a system [6].

5.7.1 Simulation Output Data Types

OPNET simulations are capable of producing many types of output. Of course, because of the general programmability of process models and link models, developers are able to define their own types of output files, including text reports, proprietary binary files, etc. However, in most cases, modelers use the types of data directly supported by OPNET. These are output vectors, output scalars, and animation [6].

5.7.2 Selecting Data for Collection

Because OPNET-developed models typically contain a very large number of potential statistics and animations of interest, collection mechanisms are not active by default when a simulation is executed. Instead, OPNET provides a mechanism to explicitly activate particular statistics or animations so that they will be recorded in appropriate output files. This is accomplished by specifying a list of probes when running a simulation. Each probe indicates that a particular scalar statistic, vector statistic, or form of animation should be collected. Probe lists are defined using the Choose Results operation in the Project Editor. In addition, more advanced forms of probes can be defined in the Probe Editor [6].

5.8 Analysis

The final phase of the simulation project involves examining the results collected during simulation. Typically, much of the data collected by simulation runs is placed in output scalar and output vector files. OPNET provides basic access to this data in the Project Editor and more advanced capabilities in the Analysis Tool, which is essentially a graphing and numerical processing environment.

5.9 Measurement Tools

A network management tool is a software package that monitors the performance of a computer communication network through the collection and storage of specified system performance metrics.

5.9.1 The SNMP Protocol

In 1988 the specification for the Simple Network Management Protocol (SNMP) was issued and rapidly became the dominant network management software. It was designed primarily to integrate the management of different types of networks with a simple design that caused very little stress on the network.

SNMP operates at the application level using TCP/IP transport-level protocols so it can ignore the underlying network hardware. This means the management software uses IP, and so can control devices on any connected network: not just those attached to its physical network. The disadvantage with this is that if the IP routing is not working correctly between two devices, it's impossible to reach the target to monitor or reconfigure it [12].

There are two main elements in the SNMP architecture: the agent and the manager. It's a client-server architecture, where the agent is the server and the manager is the client. The agent is a program running in each of the monitored or managed nodes of the network. It provides an interface to all the items of their configuration. These items are stored in a data structure called a management information base (MIB). It's the server side, as long as it maintains the information being managed and waits for commands from the client.

The manager is the software that runs in the monitoring station of the network, and its role is contacting the different agents running in the network to poll for values of its internal data. It's the client side of the communication [12].

In essence, SNMP is a very simple protocol as long as all the operations it performs deal with the fetch-and-store paradigm, and this allows for a small commands set. The SNMP has the following key capabilities:

- Get - enables the manager to retrieve the MIB object values at the agent.
- Set - enables the management station to set the value of objects at the agent.
- Notify - enables an agent to notify the management station of significant events such as errors.

The extensibility of the protocol is directly related to the capability of the MIB to store new items. If a manufacturer wants to add some new commands to a device such as a router, he must add the appropriate variables to its database (MIB). Almost all manufacturers implement versions of SNMP agents in their devices: routers, hubs, operating systems, and so on [12].

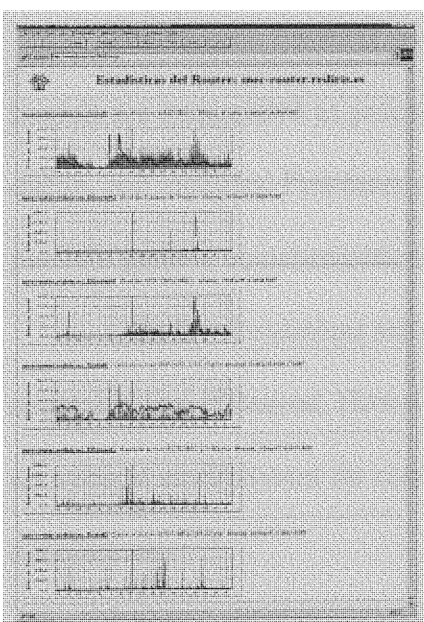
5.9.2 MRTG: Multi Router Traffic Grapher

MRTG is an advanced tool written by Tobias Oetiker and Dave Rand to graphically represent the data SNMP agents brings to SNMP managers. It generates HTML pages with GIF graphics depicting inbound and outbound traffic in network interfaces in 'almost real time'.

MRTG is highly expandability and has powerful configuration capability. MRTG can be used to monitor any SNMP variables such as traffic, error packets, system load, modem availability and others. MRTG also allows the importation of data from an external program. This ability allows it to monitor login sessions of users and other information not available through SNMP. MRTG tools allow the monitoring of router for interfaces.

MRTG permits four levels of detail for each interface: traffic in the last 24 hours, the last week, the last month and a yearly graphic. This facilitates the collection of data for statistical purpose. MRTG maintains an accumulated database with all this information with the help of a consolidation algorithm that prevents the data in the logs from consuming available disk storage space [12].

MRTG generates a main page that contains the GIF images of the daily details of every interface of a router. This is a simple but comprehensive depiction, of the performance of all routers in the network.



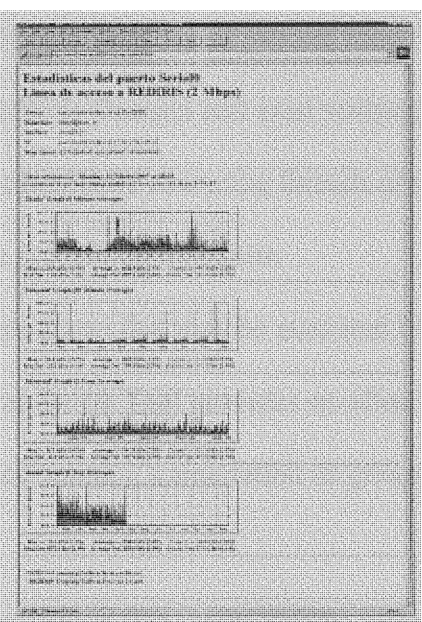


Figure 5.6a Interface Main Page

Figure 5.6b Interface Detail Page

5.10 Summary

The Project Editor is used to construct and edit the topology of a communication network model. It also provides basic simulation and analysis capabilities. The Network Domain in which the Project Editor works is the highest modeling level in OPNET in the sense that it encompasses objects which are defined in the other modeling domains. The network model therefore specifies the entire system to be simulated.

The Node Editor is used to specify the structure of device models. These device models can be instantiated as node objects in the Network Domain (such as computers, packet switches, and bridges).

The Process Editor is used to specify the behavior of process models. Process models are instantiated as processes in the Node Domain and exist within processor, queue, and modules. Processes can be independently executing threads of control that perform general communications and data processing functions.

OPNET simulations are ordinary programs that may be executed from the command line if desired. However, the Simulation Tool provides an often more convenient environment for configuring and running a simulation or group of simulations.

OPNET simulations store statistical data in two types of files, called output vector and output scalar files. Output vector files contain dynamic time-series data showing how quantities change over time during a single simulation run. In an output scalar file, each variable is usually represented once per simulation. The tool provided by OPNET for advanced processing of both types of simulation output is called the Analysis Tool.

SNMP is a simple but powerful protocol that can help us monitor our resources with little stress to the network. MRTG is a network management software that is based on the SNMP protocol. It facilitates monitoring of network objects (nodes) through a graphical display of network parameters that are represented by MIB object values. The graphical displays are on html pages using GIF images.

Chapter 6 Design and Implementation

6.1 Building Network Topologies

The initial step towards performance analysis of a network is to baseline the network topology. The method for building the network topology varies depending on what information is to be obtained from the performance analysis. For example, a study involving application deployment may require the entire topology to be modeled explicitly. A backbone analysis may require the backbone portion to be modeled in detail but allow the remaining portions to be abstracted, a client-server performance analysis may require modeling of a single path between the client and server [13].

OPNET supports many different techniques for building network topologies. Topologies can be created manually or imported from tools that perform auto-discovery. The model library that comes with the software provides models of various devices used in current day networks. Devices that aggregate sections of the network (such as LAN segments, ATM/Frame Relay clouds) are provided in order to simplify the network topology and improve simulation performance.

There are five basic steps for building a network topology.

1. Determine Goal
2. Assess Existing Network
3. Determine if Aggregation is needed

4. Select a Building Technique
5. Create Network Topology

6.1.1 Determine Your Goal

The choice of devices for constructing a network topology and the method is very much a function of a simulation study. Depending upon the context in which the software is being used, you may represent the topology as a single path between two devices, a partial topology with portions of the network abstracted or a complete topology with every device explicitly modeled.

6.1.2 Assess The Existing Network

Before beginning construction of the final topology, it is important to have a complete picture of the existing network in hand in the form of a drawing or a map. It is important to identify all the existing devices, their role and the protocols that are running in the existing network. It is also important to identify the flows and traffic patterns in the existing network. Consideration of the following questions is important:

1. Is the network flat (predominately switched) or segmented (routed)?
2. Where are the main servers located (WWW server, database server etc.)?
3. What is main traffic flow (users accessing the database server, traffic through a firewall etc.)?
4. What are the sources of broadcast and multicast traffic?

6.1.3 Determine If Aggregation Is Needed

Before performing the final steps in generating a topology, it is important to determine if you can aggregate portions of the network. Aggregation can be performed at the segment level (LAN segment) or at the subnet level (IP subnet). Portions of the network, which are outside the corporate control such as the Internet or the carriers, can also be represented as simple "cloud" objects with the appropriate latencies.

6.1.4 Select a Technique for Creating Topology

Topologies can be created manually or automatically through a process of importing from tools that perform auto-discovery.

6.1.4.1 Direct Import

OPNET supports importing topologies directly from tools such as HP Network Node Manager. Some key features of the import process are:

- The import preserves the network layout and hierarchy. The relative positioning of objects is preserved. If objects are within subnets, the software will create subnets and place objects within them.
- Devices are mapped accurately to the model library. The software maintains a large database of device models(e.g. routers, switches, servers etc.) and their

characteristics. During import, devices are identified based on their function and also identified as vendor specific.

- A Question/Answer database provides a method of dealing with unmanaged devices.

6.1.4.2 Manual Construction

Manual construction can be used when the topology is simple (in terms of objects and complexity of interconnection) or when supporting tools that provide topology information for direct import are not available.

In order to build topologies by hand, the software provides a number of object palettes that contain common network devices and links used for interconnection. Once a topology has been created by hand, OPNET provides features that allow the user to configure object models. Consistency checking is provided to ensure that links are accurate and the interconnected devices are compatible.

6.1.5 Create Topology

Once the steps have been completed, the model can be accurately built. Change can also be made to selected objects before simulation.

6.2 Traffic Modeling

The need for a formal and validated model of network traffic has been widely recognized by both academia and industry. At the same time, the small effort dedicated to the task of network design projects has also been recognized. The reasons offered to explain the apparent or real contradiction include:

a) Lack of the existing company's technological capacity required to monitor traffic in real network for later use in the design process.

b) Inadequate methods of collecting data in those cases where real traffic data has been systematically gathered.

c) Lack of the statistical background and tools required in analyzing the data by traffic analysts and networking managers in those cases where data are made available.

6.2.1 Traffic Characterization

Characterization of network traffic plays a crucial role in simulation. Traffic information can potentially affect results and conclusions drawn from the model. The OPNET provides different techniques for modeling traffic. Traffic information can be input manually by configuring various application and traffic attributes. Traffic can also be imported directly from traffic collection tools or can be input in the form of ASCII data from a file.

There are two fundamental representation of traffic:

- Explicit traffic- packet-by-packet data transfer with each transfer modeled as a discrete event.
- Background traffic- modeled analytically. Impacts explicit traffic in the form of network delays.

Simulation studies may use explicit traffic only, background traffic only or a combination of both [14].

Traffic characterization is divided into five basic steps.

1. Obtain Traffic Matrix
2. Decide which flows affect performance
3. Select a Traffic type
4. Select Modeling Technique
5. Incorporate Traffic Information

6.2.2 Obtain Traffic Matrix

Traffic in a network has complex characteristics and an important step in modeling traffic is understanding and defining traffic flows. On a given network, it is useful to classify the different types of traffic, determine the direction of traffic flow and take measurements. An attempt was made to address the following questions:

1. What are the primary applications that generate traffic?
2. Can the traffic be classified in terms of protocol (TCP,UDP,IP etc.) ?

3. Where are the servers located on the network?
4. What is the primary direction of traffic (towards server, local to LAN's, crossing the backbone etc.)?
5. Is there bi-directional traffic (HTTP download, FTP transfers etc.)?
6. Is there any unaccounted traffic such as nightly backups, email replication etc.?

6.2.3 Decide Which Flows Affect Performance

It is important to understand what is the goal of the simulation and decide which traffic flows should be modeled and which can be ignored safely. For a backbone analysis, it is important to capture all flows going across the backbone only. Local LAN segment traffic that do not traverse the backbone can be ignored. In modeling traffic all the possible flows in the network were listed and the important ones were included in the model.

6.2.4 Select Traffic type

Selection of a traffic type involves understanding the traffic patterns that OPNET allows a modeler/network designer to specify.

The figure above demonstrates traffic flows in a network that can be represented by background traffic. The profile of a specific flow specifies how many packets/sec and bits/sec were transmitted as a function of time. Traffic is automatically routed based on the network topology and the routing protocols.

6.2.5 Select Modeling Technique

In OPNET traffic information may be configured manually or can be imported directly from tools that measures live network traffic.

- Importing Traffic- Tools such as HP Netmetrix or the Network General Sniffer generally capture traffic flows between source-destination pairs. Protocol analyzers can also be used to capture all packets in the network and provide a line-by-line decode of each packet.
- Manual Traffic- Attributes on devices can be edited manually to generate traffic. Background traffic can be configured on devices and links to represent resource usage. Explicit application traffic can be configured manually by specifying request and response rates, request and response sizes and their relevant distributions. For example, to represent FTP traffic, you can specify the number of files transferred in an hour and also the average file size [14].

6.2.6 Incorporate Traffic Information

Once the traffic specification is complete, it is important to ensure that network traffic has been accurately replicated in the simulation. OPNET has a number of statistics that allows you to verify the generated traffic matches the expected amount.

6.3 Implementation

6.3.1 FIU ATM Backbone

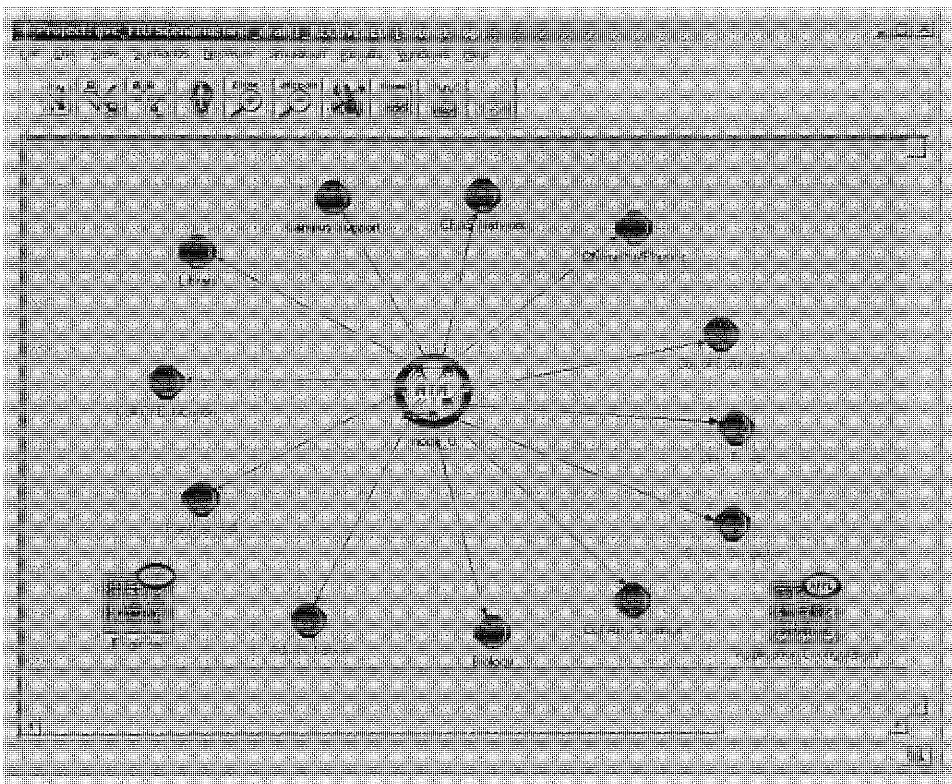


Figure 6.1 FIU ATM Backbone simulation model

Figure above is a partial topology for the FIU main campus ATM backbone. The model consist of 12 aggregated switch LANs and an ATM backbone. The purpose of this backbone model is to simulate and analyze traffic on the ATM backbone. The necessary data on network devices were collected to get information on each LAN in the model (See Appendix). Because of the purpose of the model the traffic traversing the LANs was ignored. The traffic was modeled explicitly through user applications. These are:

- Remote Login
- Email
- Web-surfing
- Database Query
- FTP
- Print

ATM cloud model was sufficient for this analysis since LAN details are not necessary. The modeled was configured by the value for its latency. The latency was determined from the following equation.

Latency = (Ping Response Time – (Hop Count * (Ping Packet Size/Data Rate))) / 2

The Ping response time and ping packet size were obtained by 'pinging' various nodes across the backbone and recording the value. The Ping command was used for this purpose. The Hop count was obtained by using the 'trace route' command.

6.3.2 College of Engineering and Applied Science (CEAS) LAN

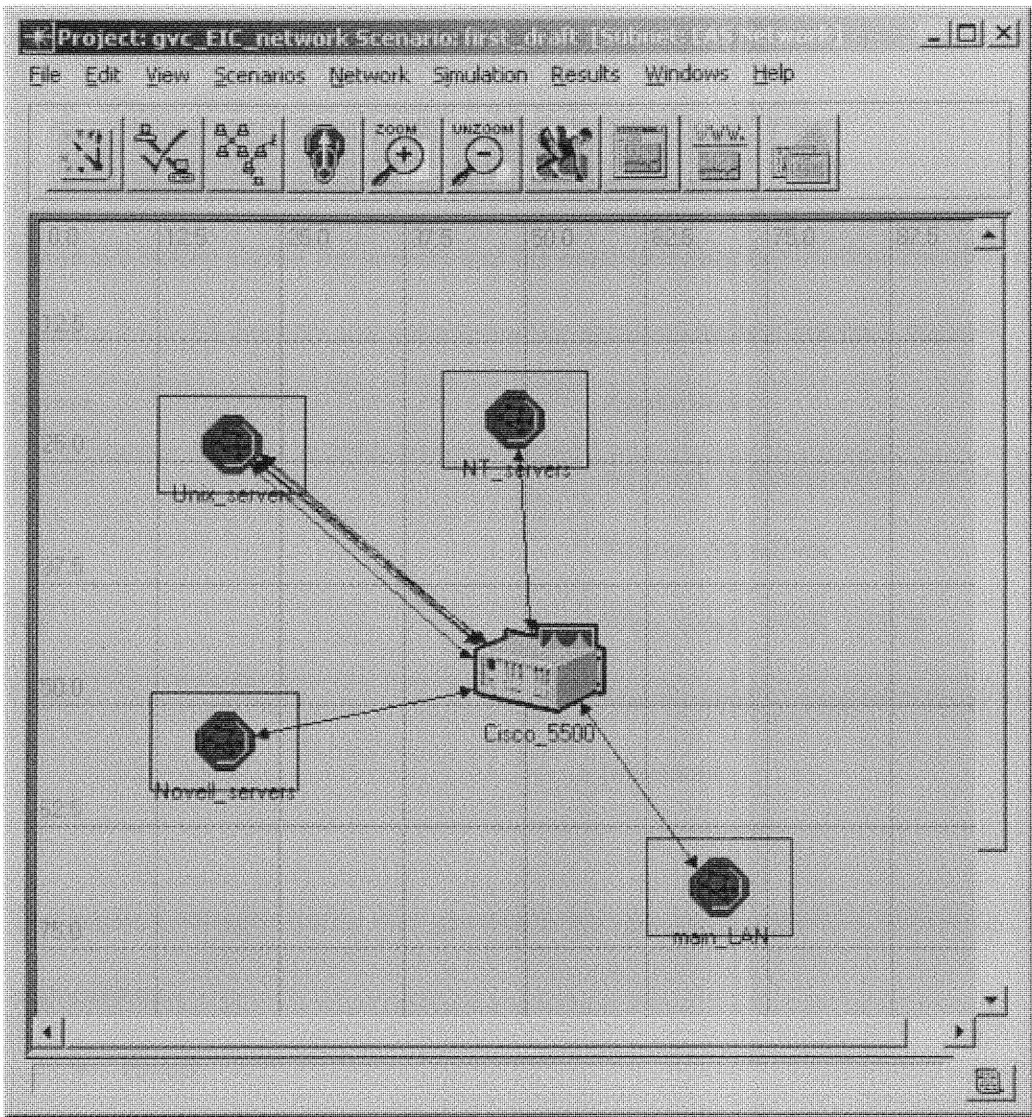


Figure 6.2 CEAS LAN

Figure above shows the highest level in the hierarchical simulated model of the EAS network. It consists of the following:

- 10 Novell servers
- 9 NT servers
- 6 Unix servers

- A total of 40 3Com switches (models 1100 & 3300)
- Cisco 5500 core switch
- Cisco 7500 router connects EAS with the main campus
- Capacity to support 960 networked workstations

Table 6.1 below shows the servers and the corresponding application that they support.

Table 6.1 CEAS Servers and corresponding application/function they support

Server	Application
Novell	Network authentication, Home directory, Printer, file server
NT	Web, email, DNS, DHCP
Unix	Mainly used to support simulation programs

The explicit network traffic was generated through modeling of the following applications:

- Email
- Web-surfing
- Database query
- Print
- FTP
- Remote Login

6.4 Statistics Collected

6.4.1 FIU Network

CEAS - Primary (ATM Line 2) - Max 155Mbps

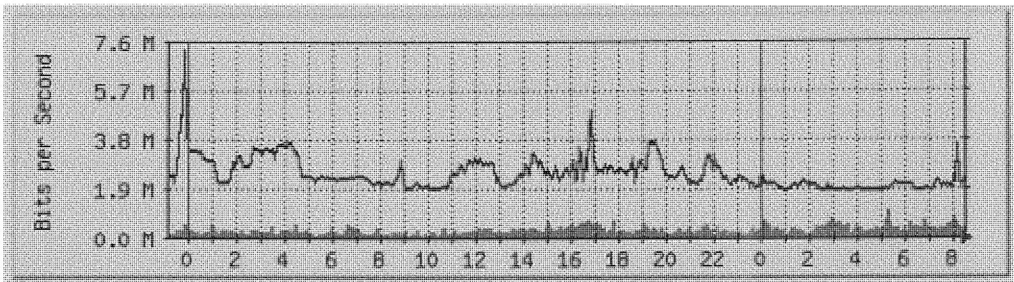


Figure 6.3 CEAS Primary (ATM Line 2)

Panther Hall Residence halls - ATM Line 1 - Max 155Mbps

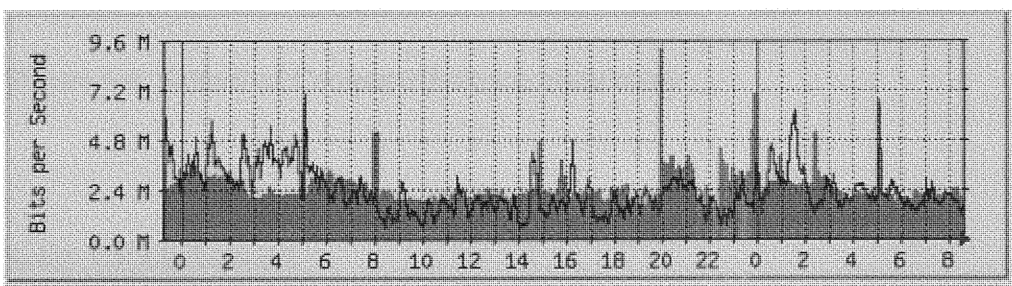


Figure 6.4 Panther Hall- ATM Line 1

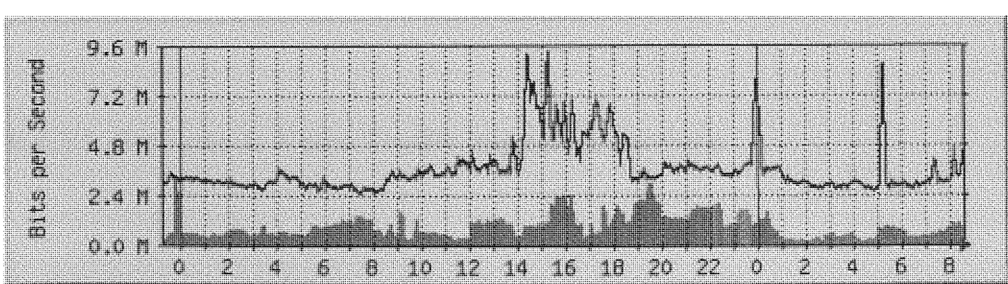


Figure 6.5 School of Computer Science- ATM line

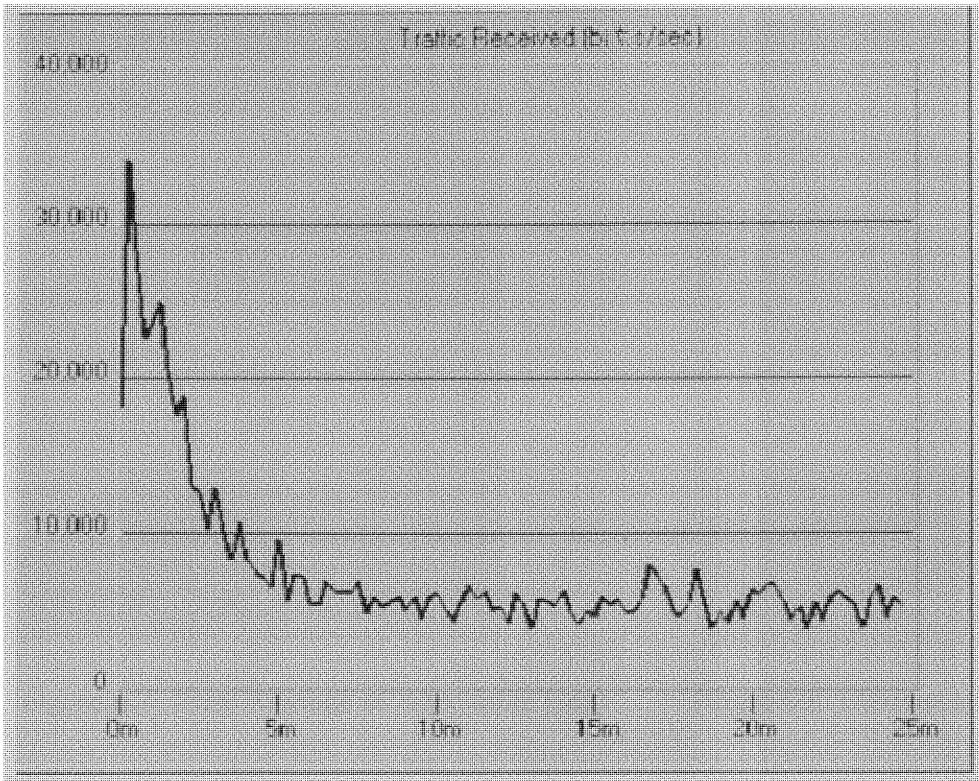


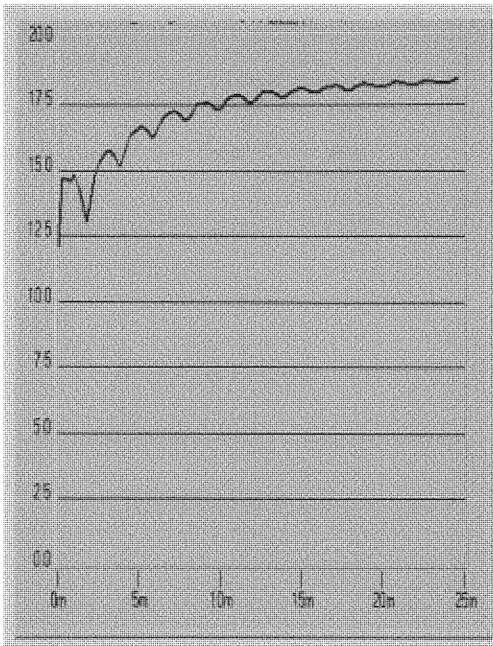
Figure 6.6 School of Computer Science- simulation

6.4.2 CEAS Network

Statistics from simulations representing three different simulation studies were collected.

6.4.2.1 Scenario 1: Voice Traffic impact on network

In this scenario, simulations with and without voice traffic on network were compared. The voice traffic represented 20% of the overall traffic on the network



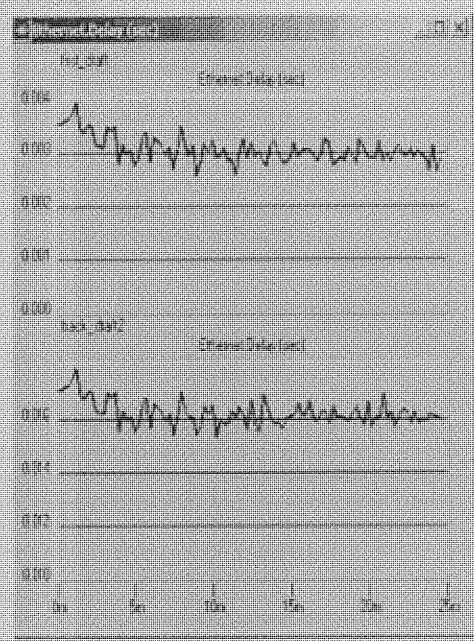
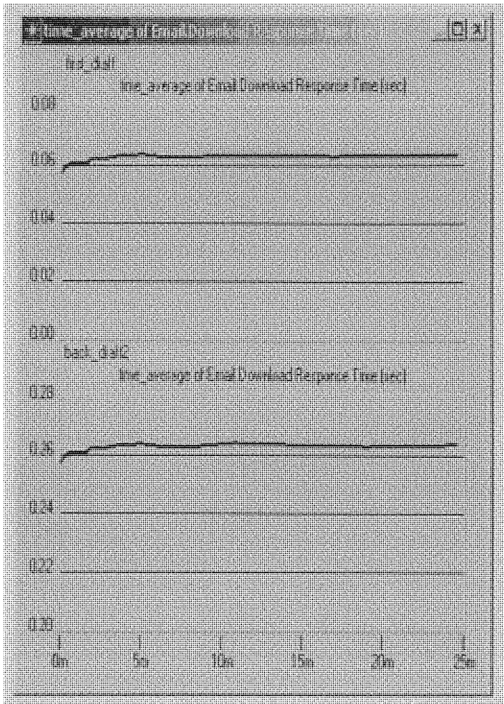


Figure 6.7 Time Average of Jitter delay (sec)

Figure 6.8 Ethernet delay (sec)

Figure 6.7 shows the jitter delay when voice traffic represents 5 % of the overall network traffic. Jitter (variability in delay) delay fell within required value of a few hundred milliseconds. However, this was essentially the cut off margin. Further increase in voice traffic resulted in jitter delay significantly above required.

When voice was added to the network the Ethernet delay increase by 400% (Figure 6.8). This is consistent with the fact that increase traffic would result in greater collision hence more delay in traversing the network. Also greater traffic would result in longer queuing delays at switches.



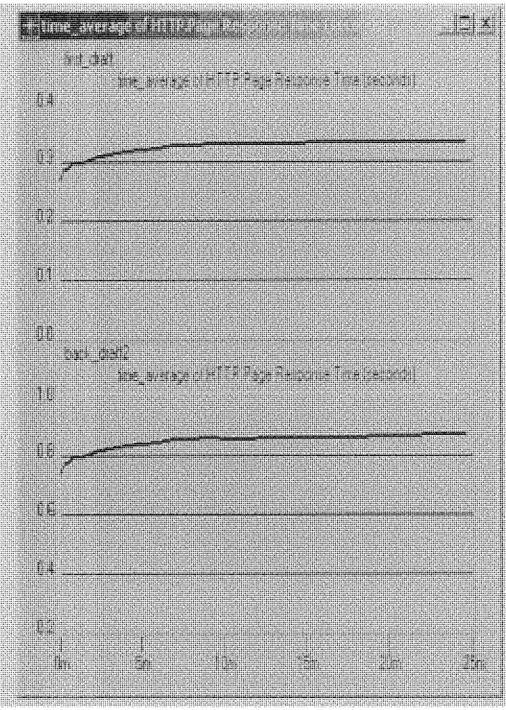


Fig.6.9 Time average of Email Download Time (sec) Fig.6.10 Time average of HTTP Page response time

Email download response time increased by approximately 433% while HTTP page response time increased by 267%. Both these figures are consistent with the fact that increased in traffic due to voice on the network will increase transmission time.

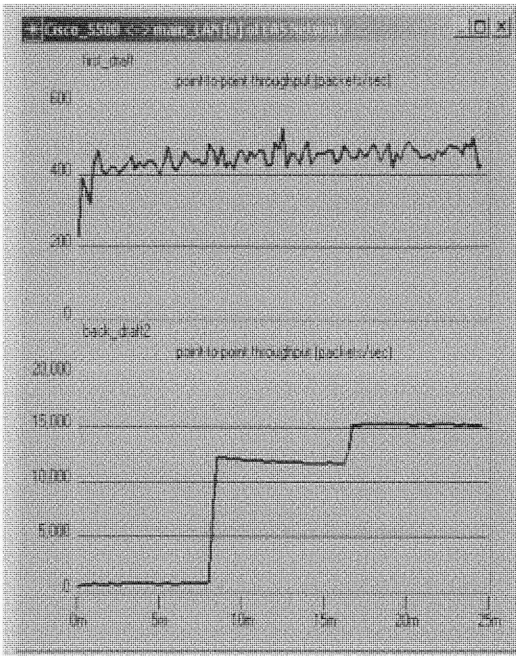


Figure 6.11 Throughput (packets/sec)

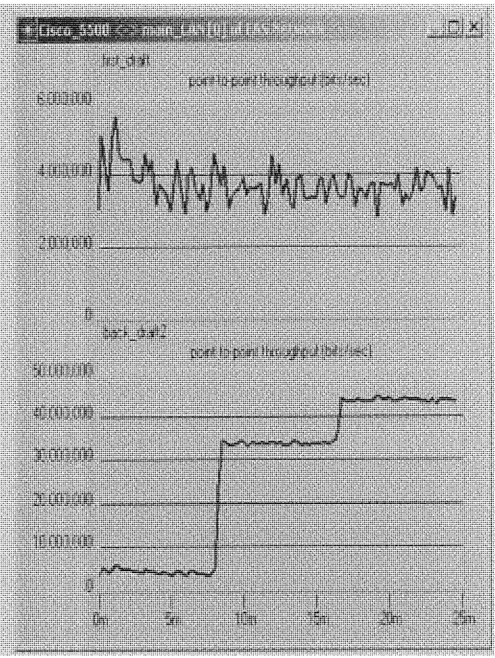
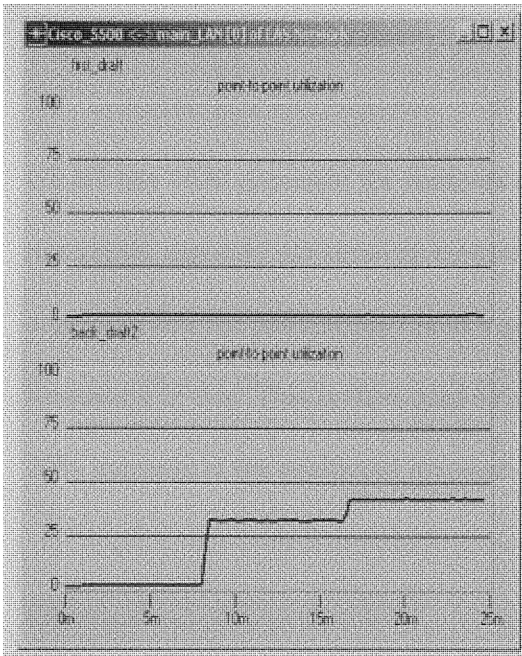


Figure 6.12 Throughput (bits/sec)

Voice addition to the network resulted in through (packets/sec) increasing by 2500% (fig. 6.11). Throughput (bits/sec) ranges from 3.0 million bits/sec to 5.5 million bits/sec (fig. 6.12). When voice was added throughput bits/sec range increased to 5-45 million bits/sec. This represents a simulation of a typical day from 8:00 a.m. to 12:00 noon. In the morning traffic is low and gradually increase to a peak as it approaches noon.



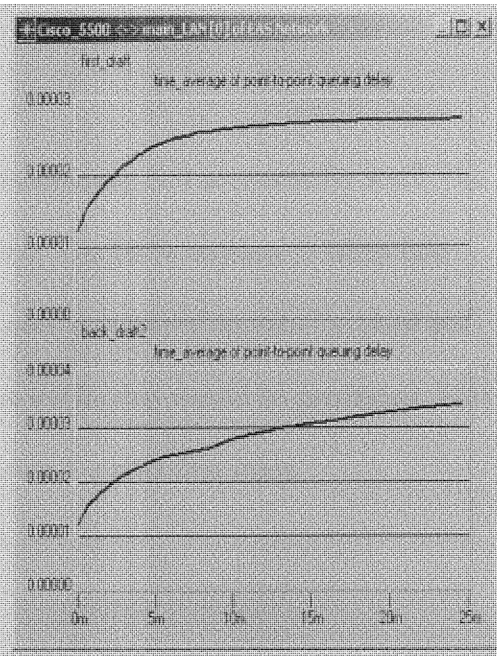


Figure 6.13 Point to Point Utilization (%) Figure 6.14 Queuing delay (sec)

Before voice the point-to-point utilization was 2% (fig. 6.13). Utilization increased to 40% when voice traffic was added to the network. This indicates a greater utilization of bandwidth on the link. However, greater utilization results in greater collision and hence greater delay on an Ethernet network employing CSMA/CD network accessing standard.

Figure 6.14 shows an increase in queuing delay when the voice is added. This is expected since the voice contributes to an overall increase in network traffic and greater utilization of bandwidth. The maximum delay is approximately 30micro seconds without voice. With voice the maximum delay is approximately 40 microseconds. This reflects a percentage increase of 33%.

6.4.2.2 Scenario 2: 20% Increase In User

CEAS has 1600 accounts for students and approximately 300 accounts for staff. The maximum capacity for network computers is 960. The 20 % increase is takes into consideration the present profile of a typically user in terms of application used and frequency of use. This is then used to model the traffic that would be generated. This 20% increase is the predicted increase in user over the next five years.

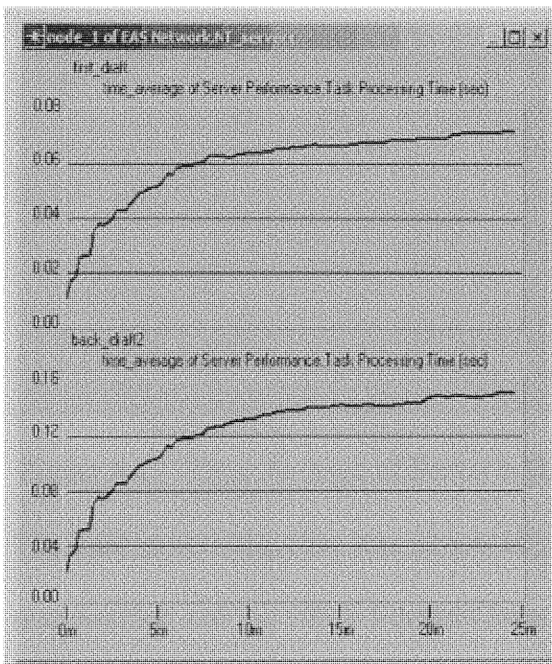


Figure 6.15 NT Server Task Processing Time (sec)

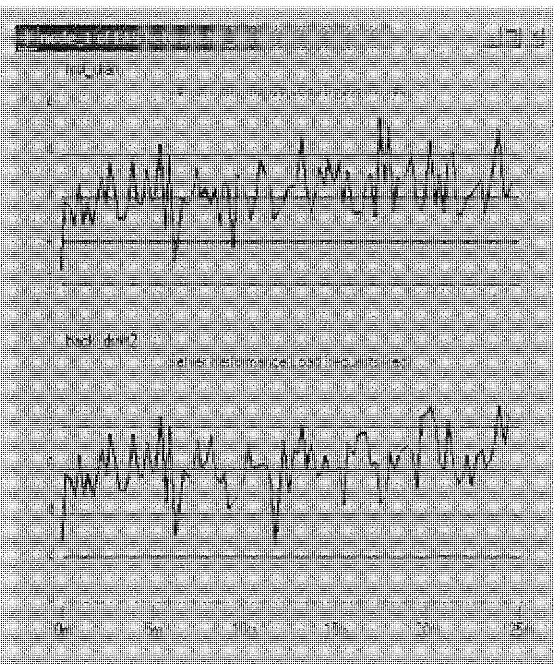


Figure 6.16 NT Server Load (requests/sec)

When the network user is increased to 20% the task processing time increased by approximately 100% (fig. 6.15) on the NT server. The corresponding increase in load (request/sec) is approximately 70% (fig. 6.16).

6.4.2.3 Scenario 3:User increase by 20%, CPU speed increase by 200%, upgrade 10Mb links to 100Mb.

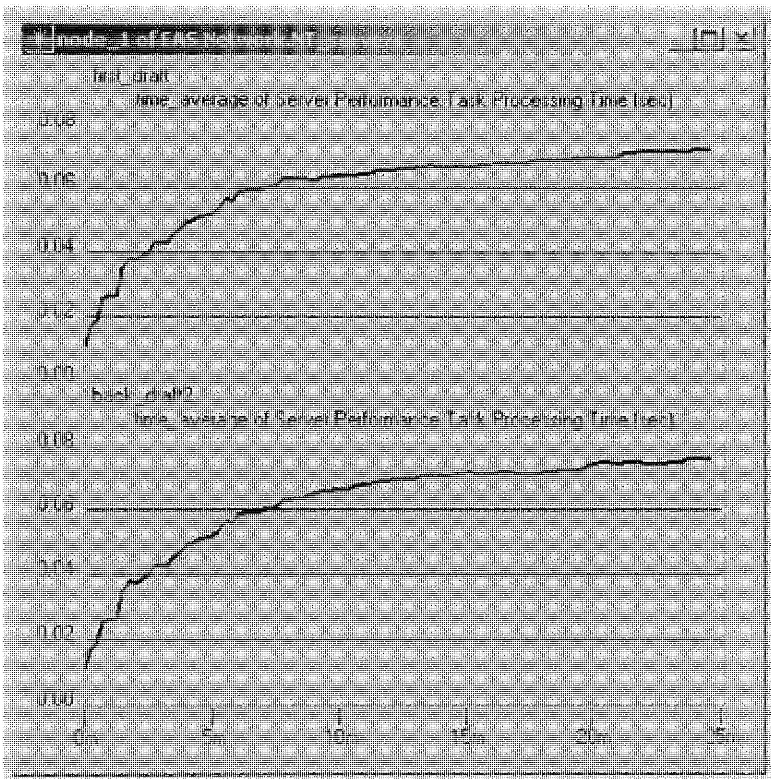


Figure 6.17 Time Average of Server's Task Processing Time

The Task processing time was almost identical to what it was before when the CPU speed was doubled and the 10Mb Ethernet links in the network was upgraded to 100Mb (fig. 6.17).

Chapter 7 Results and Conclusions

7.1 General Conclusions

I discussed findings from the modeling of an simulation of an existing campus network. Two main models were created:

- Main Campus ATM backbone
- Detailed Model of College of Engineering and Science LAN

Backbone analysis was done using the former of the two models mentioned. This model demonstrated a strong correlation with actual measurements taken using SNMP based measurement tool.

The later model was used to predict the performance of the network under different conditions. Voice was added to the network and it was observed that the network would not be able to reliable and efficiently transport voice. This was determined by the value of the jitter delay, Ethernet delay and download response time for Email and HTTP page.

The performance of the network was also studied in light of a predicted 20% increase in users of the network. Objects of the model in particular the server CPU speed and the Ethernet links was upgraded and it was determined that this resulted in a reduction of the Task processing time for the NT server. In fact, the task processing was almost identical to its value before the 20% increase in users.

7.2 Contributions of Thesis Work

1. This thesis research has provided two complete performance models of an existing campus network. Further research can be carried out through the modification and/or simulation of these models.
2. This thesis research has provided network data that were unavailable prior to this thesis.
3. This thesis research has resulted in the creation of models that can be used to predict future performance of the network in light of projected increase in user, deployment of new application and upgrade in network infrastructure.

References

[1] P.G. Harrison and N.M. Patel, Performance Modelling of Communication Networks and Computer Architectures, Addison-Wesley, 1993

[2] R.L. Sharma, P.J. Sousa and A.D. Ingle, Network Systems, New York: Van Nostrand Reinhold, 1982

[3] W. Stallings, Data & Computer Communication, New Jersey: Prentice Hall, 2000

[4] S.Wunnava, G.V.Crosby, A Kapasi, "Adaptive Network Modeling Scheme", Proceedings. IEEE SOUTHEASTCON 2001, pp. 109

[5] Ameet Kapasi, Adaptive, Secure and Network Monitored Communications, MS Thesis, Florida International University, Miami, April 2000

[6] OPNET Modeler User Manual Version 6.0, MIL3 University Drive, Washington D.C., 1999

[7] T.S. Ramteke, Networks, New Jersey: Prentice Hall, 2001

[8] D.E. Comer, Computer Networks and Internets, New Jersey: Prentice Hall, 1999

[9] Overview of FIU High-Speed Network Infrastructure, Florida International University, Miami (http://www.fiu.edu/campus.htm)

[10] J.Parker, B.Anderson, J.Hoffe, T.Murayama, and D.Ricke, Customer-Implemented Networking Campus Solution, IBM Corporation, International Technical Support Organization, January 1999.

[11] Livio Lambarelli, ATM Service Categories: The Benefits to the User, ATM Forum, 2001. (http://www.atmforum.com/pages/libraryfs1.html)

[12] David Guerrero, Network Management & Monitoring with Linux, June 1997 (http://www.david-guerrero.com/papers/snmp)

[13] OPNET Technologies, Building Network Topologies, OPNET Technologies, 2000

[14] OPNET Technologies, Representing Network Traffic, OPNET Technologies, 2000

APPENDIX A

LAN DATA SHEET

Name:...................................

Title/Position:...........................

Department:.............................

Date:....................................

1. Please specify the type of network topology e.g. ethernet bus, ethernet star, token ring etc. ....
2. The type of accessing technology/protocol employed e.g. CSMA/CD ....
3. Fill in the following table with as much information as you can provide. Please use n/a (not applicable) if a section does not apply to your network.

Network Equipment	Quantity	Model/Type	Description
Routers			
Switches			
Bridges			
Workstations			
Links	N/A		
Servers (File, HTTP, FTP)			

4. On the back of this page, please sketch a logical layout of your network.
5. In terms of network traffic, please estimate the percentage link utilization on a section of your network that you perceive to be a potential bottleneck. ....%
6. State, if possible, the typical percentage link utilization ....%
7. Please state any other information you deem pertinent in regards to the modeling of your network. (Use additional paper if necessary)

APPENDIX B

July 9, 2001,

Dear Julio Ibarra,

This letter serves to formally request permission to include information posted on your website, in particular portions of the paper entitled 'Overview of FIU's High Speed Network', in my thesis. All credits will be given in accordance with guidelines that cover such usage.

My request is based on the fact that my thesis research, entitled 'Predictive Performance Modeling and Simulation', includes aspects of FIU's Network. I also would like to remind you that I have already met with Maria Drake and you earlier this year and approval was granted to include information on FIU's network in my thesis.

I thank you for the assistance you and your staff have given me thus far and look forward to our continued cooperation.

Thank you.

Sincerely
Garth Crosby

Cc Maria Drake
Cc Dr. Subbarao Wunnava

From : Julio Ibarra <julio@vorlon.fiu.edu>

To : "Garth Crosby" <gvcrosby@hotmail.com>

CC : "Maria Rosa Drake" <maria@fiu.edu>

Subject : Re: Requesting Use of Material

Date : Sun, 15 Jul 2001 15:00:11 -0400

Reply Reply All Forward Delete Put in Folder... **Printer Friendly Version**

Garth - Please accept a positive response to your request. You may use the material on the NET web site for your thesis. Please let us know if we can be of further assistance to you.

Best of luck,
Julio