Planning for migration to a Next Generation Network

Eirik Hovland

A dissertation submitted to the University of Dublin in partial fulfilment of the requirements for the degree of Master of Science in Computer Science

September 2001

Declaration

I declare that the work described in this dissertation is, except where otherwise stated, entirely my own work and has not been submitted as an exercise for a degree at this or any other university.

Signed: ____________________

Date: September 17, 2001

Permission to lend and/or copy

I agree that Trinity College Library may lend or copy this dissertation upon request.

Signed: ____________________

Date: September 17, 2001

Acknowledgements

I would like to thank my supervisor, Meriel Huggard, for her enthusiasm and guidance throughout the course of this project. I would also like to thank the people at Eircom involved in this project for their assistance. Finally a word of appreciation to my classmates and friends for their friendship during the year.

Abstract

This thesis describes the development of a software planning tool to be used in the migration process from a Public Switched Telephone Network to a Next Generation Network. The tool calculates the benefits obtained by moving to an NGN in terms of bandwidth gain along with network infrastructure optimisations. The prototype created will assist network planners in the dimensioning of network links, taking into account the characteristics of different types of traffic such as CBR voice, VBR voice and video. It is assumed that ATM is the underlying carrier technology. Evaluation of the results shows that there are substantial savings to be obtained by investing in the implementation of an NGN.

Contents

List of Figures

Chapter 1

Introduction

This thesis describes the work carried out to create a software planning tool for migration to NGN. The project was proposed and supported by Eircom, and was submitted as an exercise for an M.Sc. degree in computer science at Trinity College, Dublin.

The immense growth in popularity of multimedia applications calls for a change in the way we think about communication networks. Next Generation Networks lay the foundation for future communication solutions by integrating data traffic and traditional telephone traffic. To allow this to happen, fundamental changes to today's communication systems are required. This work covers an area seemingly lacking previous research, namely that of analysing aspects relating to migration from today's vertically integrated networks to tomorrow's multi-service networks. The change is already upon us. Equipment vendors and network operators work together to roll out the infrastructure of a new era in communication.

Before undertaking the massive task of implementing an NGN network, operators must carefully consider whether such an operation can be justified. Until now there has been little work in the area of quantifying the benefits to be obtained by migrating to an NGN. During the migration phase, traditional PSTN services will have to be preserved while data communication services are introduced in the same network. Different traffic types have significantly different characteristics impacting the network performance, and hence the dimensioning of the network. The focus of this project is on estimating the bandwidth requirements for different mixes of traffic given certain network scenarios, as well analysing the bandwidth savings. Results from this work will assist network planners in dimensioning their network in a time when significant changes are occurring to both the network infrastructure and services offered. In addition, results from this work can provide useful information to justify large-scale investments in the rolling out a Next Generation Network.

Traffic modelling has been a main challenge of this project. Much literature is available in the area of traffic source modelling. However, finding an appropriate model has required extensive research. Assumptions were necessary because of no prior experience with integrated networks dimensioning. Other challenges in this project include finding a uniform way to measure and represent traffic in scenarios with a mix of different types of network traffic.

The project achieves to develop a flexible software tool that is capable of handling any traffic type the user defines. The tool also gives the opportunity to manipulate network scenarios, analysing the impact on the network. The flexibility of the final product is such that it is suitable as a platform for further development.

1.1 Project goal

The lack of planning and dimensioning techniques for NGNs is the concern of this project. The goal is to create a software tool to be used by network planners during the migration phase from a PSTN to an NGN. Specifically, the project aims to create a tool that estimates required bandwidth with a high level of accuracy. Another aspect is to ease the work of network planners by automating the process of calculating network capacity requirements. Network scenarios, including topology and traffic type settings, will be stored in a database. While the behaviour of the network for different traffic matrices can be analysed, an intuitive graphical user interface assists in the operation of the tool.

1.2 Project scope

The project is limited to creating a planning tool for the migration phase from PSTN to NGN. It is not the intention to create a PSTN planning tool or an NGN planning tool, but to produce useful results that can assist planners in making decisions regarding the implementation of an NGN. The tool takes into account a few important types of traffic to demonstrate their significant impact on network dimensioning. However, the traffic models are very general. It is possible to model the network traffic far more accurately, but that is outside the scope of this project. It is more important to leave room for flexibility rather than achieving high accuracy, and focus more on automating an otherwise time-consuming task.

1.3 The business aspect

There is no doubt that an NGN will lead to increased profitability in the long run. NGN solutions are expected to cut costs in various areas. Planning, maintenance and management costs will be reduced substantially by the simple fact that the complexity of the network infrastructure will decrease. Equipment costs will be reduced as network equipment becomes more uniform. Finally, more efficient utilisation of bandwidth will also contribute to higher cost-efficiency. However, migrating to an NGN is not a simple task. Careful business modelling is necessary in advance of such an important undertaking. The article [Swe01] sheds some light on things to be considered carefully for any business interested in NGN technology. The decision to invest in NGN should be based on a good understanding of the company's market position. If NGN implementation does not lead to increased profitability, there is no point in investing time and money in it. One strong argument for deciding in favour of an NGN is the intangible benefit of an enhanced enterprise image. Another pro-NGN argument is the promise of shorter time to market for new applications and services. This is very important in today's ever faster growing and highly dynamic market. Flexibility is also an attractive feature of NGN technology. The NGN concept is called future proof, meaning it should be able to handle not only today's communication services, but also those of the future. However, technically an improvement and looking very promising, other sides of the business aspects in relation to NGN must be considered. The bulk part of the revenue for a telecom operator such as Eircom is generated by its existing PSTN network services. It is extremely important to preserve all current services so that revenue is not lost or jeopardised due to implementation of a new network infrastructure. This is the reason why NGN must be phased in as a long process. The risks are many and cannot be neglected, so implementing an NGN is a process of many steps. One of which is the development of the planning tool described in this thesis.

1.4 Thesis outline

The rest of the document is outlined as follows: Chapter 2 covers Next Generation Networks state of the art, and reviews research related to this project. A brief introduction to ATM technology is given in chapter 4, along with a discussion on CAC issues. Chapter 5 describes the design of the software tool, and chapter 6 covers the implementation process. Collection and analysis of results are detailed in chapter 7, and finally chapter 8 calls for further work before giving some final remarks.

Chapter 2

State of the Art

This chapter will briefly cover the current state of NGN technology, and also discuss a few NGN commercial products from some of the major equipment vendors in the market

2.1 NGN technologies

In the literature, [Don00] among others, mainly two technologies are held as candidates for serving as the transport mechanism for NGN. Namely, IP and ATM.

2.1.1 IP based NGN

IP is believed by many to be the choice of the future as the NGN carrier. The volume of data communication is already superseding the volume of voice communication. The advantage of IP is that it was specifically designed for handling data communication. IP's layered architecture and variable packet length makes it a very flexible transport protocol. IP possesses the ability to carry most, if not all, existing communication services, and it is common belief that IP will indeed be able to cater for future communication needs. The emerging of IPv6 will further strengthen the IP protocol's position by enhancing QoS guarantees.

However, despite being the choice of the future, IP does have a few drawbacks that must be eliminated before it can become the preferred NGN carrier. IP originated from the need to transport data correctly over an unreliable infrastructure. IP's robust packet switching is ideal for this purpose. In the face of congestion or broken network links, IP finds a way around the problem, and delivers the packets to their destinations. This is a positive feature in a data communication scenario, but where timing between sender and receiver is an issue, IP is still inferior to ATM, even if protocols like RSVP gives IP the ability to reserve end-to-end capacity in a network. Much research is being done in the area of making IP suitable for real-time services [JVG99], and some believe that IP in the near future will close the gap on ATM as the preferred NGN carrier.

2.1.2 ATM based NGN

There is consensus in the literature that ATM is still the preferred technology among network operators for carrying NGN. Telecom operators are the main promoters of this view, whereas data network operators tend to prefer IP. From a telecom operators perspective, it is extremely important to preserve current services offered to the customer. The motivation behind the design of ATM was to meet the needs of voice telephone traffic. Being a quite mature technology, ATM standards have evolved and optimised and have gained wide acceptance. ATM establishes a virtual circuit between the communicating parties making sure of sufficient network capacity on all elements of the path, thus guaranteeing a certain amount of bandwidth available to the application.

The connection oriented transport also guarantees QoS parameters, making it an ideal carrier for real-time services like telephone traffic. Furthermore, the ATM protocol, like IP, is also able to handle a wide range of different traffic types. This is enabled through ATM Adaptation Layers. These will be discussed in more detail in section 4.1.6.

ATM's fixed cell size is ideal for voice transmission, but can be a drawback for other services. For a description of this problem, see chapter 4. 53 byte cells speed up the processing of ATM traffic, but introduces unnecessary overhead. This is particularly the case when carrying IP services over ATM. In [ea98a] the performance issue of carrying IP over ATM is discussed and recent research is reviewed.

To conclude this discussion: IP will be the way of the future. ATM is the preferred choice of telecom operators as the carrier for NGN at the present. As the work contained in this thesis is for use by a telecom operator, it is assumed that ATM is the underlying network transport mechanism.

2.1.3 NGN solutions and implementation strategies

There are already various commercially available NGN products on the market. NGN is currently the latest in telecommunication, and all the serious players in the market want to be part of it. The major equipment vendors are planning, or already offering NGN solutions, and the operators are eager to implement them.

Ericsson offer a whole family of NGN solutions which, according to Ericsson themselves, guarantee a smooth migration towards a packet switched multi-service network. The name of the Ericsson NGN suite is ENGINE, derived from "Ericsson next generation network solution". Alcatel have their own NGN solution called Alcatel 1000 Softswitch, also promising a seamless transition to NGN.

Ericsson ENGINE

ENGINE comprises a complete package of NGN solutions [MHl00] based on ATM as the carrier technology, promising a smooth migration to an NGN

ENGINE trunked network, also referred to as the "Dynamic trunking" solution, interfaces PSTN nodes to an ATM backbone network using a so-called Media Gateway. The solution enables virtual trunks to be set up in the ATM backbone according to the load on the network.

ENGINE Bridgehead - the telephony single-domain solution. ENGINE Bridgehead uses a telephony server to provide all functionality expected from a telephone network; signalling, call routing, charging and accounting. The transport responsibility and the call control are separated into two independent layers, increases the flexibility of both. A Media gateway interfaces the circuit switched environment with the ATM core, making the interconnection of networks transparent to the PSTN network.

ENGINE switched network implements a core network capable of handling both telephone and data traffic. Telephone traffic is carried through the core ATM network by means of AAL1 64k circuit emulation, allowing dynamic resource allocation. As opposed to ENGINE Bridgehead, this solution can interconnect multiple domains.

ENGINE Integral is Ericssons full NGN solution. It preserves all existing telephone features while carrying the traffic over an ATM core network. Call set up is performed using AAL1 circuit emulation between a caller and a telephony server. Telephony servers communicate using standardised protocols tunnelled through ATM. At the edge of the ATM network, media gateways (MGW) serve as interfaces. Between the local exchanges and the

MGWs standard PSTN call setup is used. After call setup, resources are allocated across the ATM network between the participating MGW's. Again, full transparency is provided.

The ENGINE solution package is focused on retaining existing telephone services, and hence targeted at telecom operators. ENGINE future enhancements are planned to include an IP transport backbone in addition to ATM, using the same principles as the current solution.

Ericsson ENGINE products are currently available and have been put into service. As an example, MediaWays, one of the largest German ISP's have signed a contract with Ericsson to provide equipment from the ENGINE programme. The installation started in UK, and will later expand to all countries where MediaWays operate. According to MediaWays and Ericsson this solution will quickly generate revenue by offering full multimedia functionality as well as standard telephone services.

Brazilian CTBC are upgrading their network infrastructure with the Ericsson ENGINE Integral at present (Sept. 2001). The conversion is expected to increase the capacity of the current network and offer new services to the customers, making CTBC pioneers on the South-American telecom market.

ENGINE solutions are also being deployed in Spain and other European countries. Eircom are also looking at Ericsson NGN solutions for implementation over the next few years, hoping to become a technology leader in a highly competitive market. These are just a few examples to emphasise that NGN solutions are already being implemented.

Alcatel Softswitch

As an important telecom equipment vendor, Alcatel have their own NGN solution package; Alcatel Softswitch. Like the Ericsson counterpart, the

Alcatel Softswitch promises seamless migration from PSTN to NGN. The report [Tek01] gives a brief overview of the migration. The customer can either implement full NGN functionality in one step, or gradually expand the functionality of the current network. In either case, Alcatel guarantee preservation of existing services. The Softswitch solution suite is currently available.

Norwegian Telenor have entered a pilot programme where Alcatel provide Softswitch products. Depending of the outcome of the pilot programme, Telenor will decide on their strategy towards Next Generation Network.

2.2 NGN planning

In this section migration paths towards an NGN will be dealt with. The report [Don00] gives an introduction to Eircom's view of the migration from a PSTN to an NGN. According to this report, a key concern is to promote Eircom as a technology leader. Eircom already have an ATM network in place, which will be the foundation for an Eircom NGN. Initially, the core ATM network will handle all the traffic. The trend is that network operators want to move away from IP- or ATM-only solutions. Therefore, at the next stage, the core connectivity network will implement both IP and ATM platforms side by side, retaining the currently offered services. In this phase bulk IP will be serviced using the IP functionality, whereas IP services requiring QoS guarantees will run on top of an ATM layer. ATM will also carry all services other than IP. The migration process will start with a set of core network nodes implementing the NGN functionality. Later, new nodes will be progressively connected to the core connectivity network. At some stage it will be economical to connect local voice switches directly to the core network.

Call control handling can then move from local exchanges to centralised call servers, eliminating the need for telephone transit switches.

2.3 Chapter summary

This chapter has covered the essential principles of Next Generation Network and given an overview of a few of the commercial products currently available. However, many other NGN products are either available or under development. Common for many solutions is that they try to make the transition from the current technology to an NGN as seamless as possible. The focus of this project is the transition from a Public Switched Telephone Network to an NGN. The following chapters give an introduction to the relevant technologies, in this case PSTN and ATM.

Chapter 3

The PSTN network

The Public Switched Telephone Network will be migrated to a Next Generation Network over the next few years. It is important that existing PSTN services are preserved in the migration phase. Therefore, the design of a migration planning tool requires an understanding of PSTN technology.

3.1 Network structure

3.1.1 Switches

The PSTN is built as a hierarchical architecture. Closest to the telephone in the customers house, the cable connects to a local exchange or Private Branch Exchange (PBX). The local exchanges are also referred to as primary switches. The primary switches connect residential customers to the public telephone network, and handle the calls originating and terminating in the local area. Primary switches may be interconnected, in which case, traffic can be routed directly. Otherwise, calls terminating in a different local area, are sent up the hierarchy to a secondary switch.

Secondary switches handle traffic originating and terminating in different

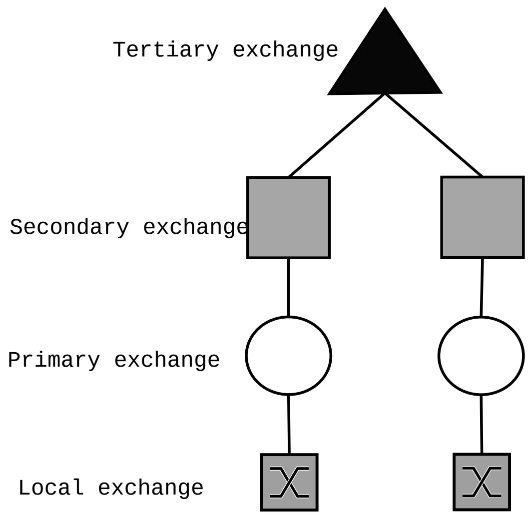


Figure 3.1: The PSTN hierarchy

local networks, but also serve as the network entry point for larger customers or for residential customers in the vicinity of the switch. The secondary switches also connect to the tertiary switches, but avoid routing traffic to the tertiary switches whenever possible.

The tertiary switches are the top layer of the hierarchy. In Ireland there are only four tertiary switches. For security and load balancing purposes they operate as two pairs. The tertiary switches handle traffic between secondary switches that are not interconnected. They are also the connecting point to other networks, like the mobile network and international networks.

3.1.2 Routes

Routes between switches in the PSTN hierarchy can be classified as high-loss routes or final-choice routes. Where large volumes of traffic originate and terminate between two switches, direct routes may be provisioned. Such routes can connect any two switches, regardless of their position in the hierarchy. These are called high-loss routes because they are dimensioned with a relatively high blocking probability. They reduce the load on the network switching nodes by bypassing some of the intermediate switches.

As the name suggests, the likelihood of congestion on high-loss routes is relatively high. The routing algorithm will look for the shortest route between two nodes. If the first option is congested, a second option with higher blocking probability will be tried. Ultimately, final-choice routes with a significantly lower probability of congestion will be chosen. Final choice routes are often sub-optimal with regards to routing. While high-loss links are provided for convenience, final-choice routes must carry the responsibility to guarantee the overall network's grade of service.

3.1.3 PSTN planning

The telephone has been a part of life for almost a century. Thus, planning of Public Switched Telephone Networks is a mature science.

Many factors come into play when dimensioning a PSTN network. Future planning is often done for three to five year intervals. A key factor in network planning is forecasting traffic demand. Traffic growth is tied to expected growth in customer base. Customer base growth is closely related to political and environmental issues. If a new residential estate is being planned, the customer base for a certain area can be expected to increase. The construction of a new corporate building will likewise contribute to growth in the number of customers. After predicting the increase or decrease in number of customers for the planning period, various techniques can be applied to forecast the traffic matrix for the planning date. A collection of such techniques are described in [She80]. When predicting future traffic volumes, certain assumptions must be made about the nature of the traffic. Research has revealed that for a short- to medium-length period, 5 to 7 years, calling rates of the customers in an urban telephone network may be seen as constant. Originating traffic and terminating traffic at network switches have the same property. The report [She80] traffic discusses traffic forecasting techniques. Some forecast methods distribute forecast traffic only taking into account either originating or terminating traffic. Such methods do not yield the constant calling rate assumption, and are therefore judged invalid for application on an urban telecom network. The Kruithoff algorithm, however, keeps both originating and terminating traffic at each switch close to constant, and can therefore be applied. The Kruithoff is also called the double factor transform method. Another algorithm, called Quadratic model also produces usable results. Since the Quadratic model is more computationally expensive, the Kruithoff algorithm is the forecasting technique used for Eircom PSTN planning. The Kruithoff method for traffic matrix forecasting is called a double factor transform method because it distributes forecast date traffic in proportion to both originating and terminating traffic. Future sums of originating and terminating traffic for each switch are computed, and then summed up. After summation of row totals and column totals, there will be a slight difference between the future sums. Before Kruithoff is applied, the sums of rows and columns must then be adjusted so the totals sums are equal. After future originating and terminating traffic for all exchanges has been computed, the Kruithoff algorithm multiplies all row elements with a growth factor, making the actual sum of the rows equal to the forecasted traffic values. Then, the new sums for all columns are computed. If the difference between column sums and the forecasted terminating traffic values exceeds a specific tolerance, all element of the column are multiplied with a correction factor. The same is then done for the rows. The last two steps of the algorithm are performed until the sums of all columns and all rows are within the tolerance of the forecasted values.

Given a certain traffic matrix, PSTN routes are dimensioned based on a blocking probability factor [Tel98]. The blocking probability is the probability of a call getting the busy signal. For high-loss routes a blocking probability of 0.05 is common, whereas 0.005 is common on final-choice routes. The Erlang formulas can now be applied to compute how many trunks are needed to accommodate a customer base of a certain size. Traffic matrices are recorded in the most busy hour of the network during the day, week and month, so the dimensioning is done for the worst-case scenario. The ratio between the average duration of a call and the call rate also has an effect on dimensioning the switching equipment. The same is expected of customer behaviour in the event of blocking. If a blocked caller repeatedly tries to establish a new connection, the network is only going to get further congested, hence degrading the grade of service. There are many aspects to PSTN planning, but this report does not intend to discuss this matter in further detail. For more information on PSTN planning, [Tel98] provides extensive explanations.

Chapter 4

ATM fundamentals

An understanding of fundamental ATM issues is necessary for the design of this software. This chapter covers basic ATM features and points to literature for further reading. Special emphasis has been put on ATM traffic source modelling as well as the investigation of Connection Admission Control (CAC) algorithms commonly used in ATM. These particular areas forms the basis for many of the assumptions made in the design and implementation of this project as well as in the analysis of the results.

4.1 ATM technology

ATM broadband technology has matured over many years. It provides high-speed cell-based switching and guaranteed QoS. ATM was designed to meet the requirements of voice communication requirements and has during its lifetime been optimised for this purpose. Abilities to carry a wide range of different types of communication have been standardised in the ATM protocol. ATM serves as network transport mechanism in many corporate and private networks, and is, as mentioned in chapter 2, currently seen as the best transport mechanism for multi-service networks.

4.1.1 The ATM cell

From a data communication point of view, the fixed ATM cell size is not optimal. However, seen from a telecom perspective, fixed cell size increases efficiency and decreases the computational cost. 53 bytes was a compromise between data communication experts, who wanted larger cells, and the telecom community who would have preferred even smaller cells.

The structure of the ATM cell is shown in the figure below.

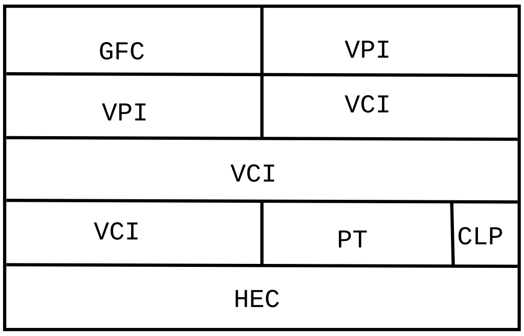


Figure 4.1: The ATM cell header

GFC - Generic Flow Control provides to ability to implement flow control mechanisms on the user side of the network. This field is 4 bits.

VCI, Virtual Channel Identifier, and **VPI**, Virtual Path Identifier, hold routing information. Together these two fields require 24 bits.

PT - Payload type indicates the nature of the content in the information field. Three bits are used for this purpose.

CLP - the Cell Loss priority bit allows for two priority levels. Cells with this bit set to zero will be discarded first in the event of network congestion

HEC - Header Error Control performs a check on the correctness of the data in the header. No error control is implemented for the payload. This is ip to the receiver of the cell, since different types of traffic require different response to errors.

4.1.2 Traffic policing

At connection setup, QoS parameters are negotiated between the application requesting service and the ATM Connection Admission Control, CAC. Different CAC algorithms are described in 4.3. For constant bit-rate services, a peak cell rate, PCR is allocated. Cells exceeding the allowed bit-rate are discarded.

Variable bit-rate resource allocation is more complex. In addition to maximum bit-rate, a sustainable cell-rate, SCR, and a maximum burst size, MBS are negotiated. SCR is the average bit-rate the application is allowed to send. Cells are discarded if the MBS or PCR are exceeded. After a burst, the source must transmit at a bit-rate lower than the SCR until the average bit-rate is back at SCR. It is the responsibility of the customer to make sure traffic complies with the negotiated policy.

4.1.3 Routing

A virtual channel is one traffic flow between two end nodes. A VP comprises of multiple VC's. VP's may be routed as an entity, of it can be split into a number of VC's. A VP on the other hand, may not be split up. VPI and VCI are relevant only to the next switch, which modifies the fields according to its own routing table, which again holds information about its neighbouring nodes.

4.1.4 ATM multiplexing gain

Statistical multiplexing gain is one of the features that contribute to making ATM so attractive. The effect of the multiplexing gain is that the bandwidth required to transmit traffic from numerous sources is less than the sum of the average bit-rates of each individual source. Characterising ATM multiplexing gain is a complex mathematical procedure, which will not be described here. The benefit of the multiplexing gain is that the more traffic multiplexed onto the same network link, the better the bandwidth utilisation. Enhanced bandwidth utilisation is always a good way to reduce transmission costs and thus, increase profitability.

4.1.5 ATM traffic types support

ATM supports a wide range of traffic types. In order not to get lost in the wilderness of traffic types, ATM classifies traffic into five different categories.

CBR (Constant Bit-Rate) for applications such as uncompressed audio or video requiring a guaranteed level of bandwidth.

VBR (Variable Bit-Rate) for high priority traffic with delay and jitter restrictions such as compressed video and audio streaming. Bandwidth allocation is negotiated and guaranteed according to Peak cell rate, sustainable cell rate and burstlength, etc. Non-complying traffic is subject to loss.

VBR-rt (VBR-Real Time) for real time applications such as video conferencing and compressed Voice over IP telephone calls.

UBR (Unspecified Bit-Rate) for file transfer, and other data communication applications insensitive to delay and jitter, but with strict no loss requirements.

ABR (Available Bit-Rate) for applications which do not tolerate packet loss, but are insensitive to delay and jitter. ABR is prioritised higher than UBR.

4.1.6 ATM Adaptation Layers

The implementation of different AALs (ATM Adaptation Layer) gives ATM the ability to carry a wide range of different traffic types. The AAL adapts a traffic stream to ATM standard, so that it may be carried over the ATM network. An example of an AAL task would be to break large IP packets into 48 byte chunks of data and put them in ATM cells. This section outlines the purpose of each AAL. [dP] gives a more detailed discussion on the issue of AAL, but lacking description of AAL5 which was developed after the report was written.

AAL1 is intended to serve CBR real-time applications. This AAL is often used as 64kb circuit emulation for PSTN voice. It is connection oriented and guarantees delay and jitter. Using AAL1, 9 of the 53 byte ATM cell contains overhead information.

AAL2 is like AAL1 connection oriented, and also provides QoS guarantees.

The main difference is that AAL2 is designed to accommodate VBR instead of CBR. AAL2 is normally used with real-time applications such as MPEG, requiring bandwidth guarantees. AAL2 gives 45 bytes of payload.

AAL3 does not give any timing guarantees between source and destination. However, it is connection oriented and designed to serve VBR traffic. 44 bytes of payload can be carried in each ATM cell using AAL3.

AAL4 has similar application areas as AAL3. The main difference is that AAL4 is connectionless. AAL4 was intended to handle IP traffic. However, efficiency proved to be very low with only 44 bytes of payload.

AAL5 was intended to take over the job of AAL4 after this not being very successful, mainly trying to enhance the efficiency of IP over ATM. AAL5 has a 47 byte payload.

Although ATM multiplexing gain enhances bandwidth efficiency, the fixed cell size of 53 bytes is not optimal for traffic types generating variable length packet. IP is such a traffic type. IP packets can be up to hundreds of bytes long. The percentage of overhead in an ATM cell is significantly higher than the overhead of IP packets, and is an addition to the total overhead when carrying IP over ATM. Another issue is that an IP packet may not fit exactly into an even number of ATM cells. In fact, in the worst case, an entire ATM cell will be needed to carry only the last byte of the IP packet, leaving 47 bytes of wasted space. In some circumstances, IP over ATM may have a throughput close to zero.

4.1.7 ATM planning

The article [Vla98] discusses different aspects of ATM network design and dimensioning, with the ASTRO (ATM Structure Optimisation) problem being a keyword. Equipment localisation and interconnection are very important to the planning of a good network infrastructure. The article also elaborates on the issue of dimensioning capacity on network links, and raises the question of what constitutes a well-dimensioned link. According to the author, poor bandwidth utilisation does not necessarily indicate the link is badly dimensioned. The CAC plays an important role in the efficiency of bandwidth utilisation, and hence on the dimensioning. The responsibility of the network planner is to assure grade of service parameters are met. Customers are encouraged not to ask for more bandwidth than they need. In the final remarks, the article [Vla98] promotes planning for flexibility rather than accuracy. This article argues in favour of the view expressed in the following section about ATM traffic source modelling.

4.2 ATM traffic source modelling

Traffic source modelling is doubtlessly the most challenging area of this project. In an ATM scenario the behaviour of the network traffic has significant impact on the level of bandwidth utilisation, and hence on the dimensioning of the network capacity. In depth studies of source traffic modelling have been conducted.

Much effort has been devoted to studying the behaviour of different network traffic types. Study of traffic is very important for the evolution of communication technology. To get the most out of current technology and to invent solutions for the future, simulation and analysis of different traffic scenarios in different network environments gives valuable knowledge. Most real life networks are in continuous operation, disallowing experimental usage. Creation of network simulators and traffic models is therefore the objective of many researchers. This is why there are so many diverse kinds of traffic models for all imaginable types of network traffic. To understand them all is virtually impossible, and nor is that the intention of this project. For the purpose of this project, research in this area has been focused on finding an appropriate way to characterise different types of traffic given the assumptions made about traffic and network in this project. Models studied are mostly Variable Bit-Rate models (VBR). Modelling Constant Bit-Rate traffic is seen as trivial from the viewpoint of this project. Unspecified Bit-Rate (UBR) and Available Bit-Rate (ABR) have not been studied for the reason that they have not been implemented in the software tool.

[HH99] provides a useful review of different traffic models. The report covers models from the simplest Markov models, to more sophisticated models that include features such as self-similarity and long-range dependence. The report concludes by classifying the different model types by their application: video modelling, Ethernet traffic, ATM traffic and so on. The classification of traffic models is supported by mathematical evaluation of each model's performance. Despite complex formulae, The report [HH99] gives a very good and extensive overview of all the research done in the area of network traffic modelling.

4.2.1 Markov models

Markov models are some of the simplest ways to generate artificial network traffic. A Markov process has a finite number of different states. For each state, associated probability factors determine the likelihood of the process to maintain the current state, or shift to any of the other states. Probabilities for all state changes are contained in a probability matrix. For traffic simulation, each state of the Markov model can represent a certain bit-rate generated by the source. State change represents a shift in the source bit-rate. The simplest Markov models consist of two states, representing the on and off states of the source. Markov models capture first order statistics like average bit-rate and peak to mean ratio well. The problem with Markov models is that the next state depends on the current state only, hence they do not exhibit characteristics such as long-range dependence and self-similarity which are often observed in real-life traffic traces.

Markov processes can also be used to modulate other source models. A Markov chain can determine shifts in the traffic stream where the parameters of the source changes. Using this approach, scene changes can be incorporated in models of VBR video streams [MRI99] or other significant changes in the traffic flow.

4.2.2 Fluid models

Instead of generating traffic on a cell level with each discrete event being the arrival of an individual cell, a fluid model can be used. It is assumed that the traffic remains constant until the flow is changed. Usually this is triggered by some event. Such changes can be controlled by a Markov process, as mentioned above. The usage of this type of model would be for ATM sources generating a close to stationary bit-rate or for large volumes of TCP/IP traffic.

4.2.3 Renewal models

Renewal models are based on estimating the frequency of events. It is assumed that all events are independent, thus the traffic observed at one instant is independent of observations made at any other instant. This means that the model does not capture autocorrelation. The application of this type of traffic model could be for scenarios where the traffic shows no autocorrelation. However, assuming no autocorrelation may give rise to unrealistic traffic models.

4.2.4 Linear stochastic models

These models are quite simple. They compute the current state depending on a certain number of previous states. As they capturing short-range dependence, they may be used to model VBR video traffic. Scene changes can be taken into account by employing a Markov modulation.

4.2.5 TES

TES is short for Transform Expand Sample. TES models capture a range of different autocorrelation functions, and is therefore suited for VBR, ethernet and web traffic. However, the model relies on traffic traces being available. TES processes can be tuned to match the specific traffic trace.

4.2.6 Self-similar models

Self-similarity is a property often found in real-life network traffic. Traffic is self-similar if it shows the same characteristics over different time scales. A wide range of self-similar models have been developed. One such model is the on/off model. Given a certain distribution of on and off times, this type of traffic model will exhibit self-similarity. Self similar models can be used to model web traffic that takes user command intervals into account.

4.2.7 Models for VBR video

[MRI99] reviews different models for VBR video. The author states that assuming independent packet arrivals is inadequate for video traffic modelling. The report classifies video models into hierarchical and non-hierarchical models. Hierarchical are those that take scene changes into account. TES, Markov and self-similar models are analysed and found to function well for different purposes. The models are evaluated in terms of complexity, number of parameters required and what level they operate on, for example cell level, frame level, etc. The report calls for further research in the area of measuring the users perception of quality. It also concludes that there is a need for more general video traffic models.

4.2.8 Traffic models discussion

As already mentioned, there is a vast number of different traffic models that suit different purposes. For the application designed in this project, one should keep in mind that the scenarios being planned do not exist at the present. [MRI99] comments that there is a lack of general video traffic models. The same can be argued for other types of traffic models, making it difficult to deem one model more appropriate than any other. For a planning tool it is more important to give a high-level overview than model different features down to the smallest detail.

[Ryu96] questions the validity of the myth that long-term correlation has a significant impact on cell loss rate in ATM traffic. Based on studies of real-life traffic, the paper proves that long-term correlation does not affect the dimensioning of realistic-size ATM buffers, and thus on bandwidth. The findings of this report and the issues discussed in section 4.3, support the view that coarse parameter-based traffic models will be appropriate to achieve satisfactory results. The parameters used for traffic type declaration are described in chapter 5.

4.3 CAC issues

A prime concern in any network is traffic control. In ATM, traffic control mechanisms are called Connection Admission Control (CAC) algorithms. CACs are a proactive approach to avoiding network congestion. There are many types of CAC to choose from, each addressing different problems, but the main idea is to limit the number of new connections admitted to the network, so that the quality of ongoing connections is maintained.

This section discusses various concerns regarding Connection Admission Control (CAC) schemes and their effect on the bandwidth utilisation in a network. The discussion concludes that an aggregated effective bandwidth approach is appropriate for this application, even if such an approach tends to be overly restrictive in bandwidth allocation. The report [Tia99] gives a more extensive coverage of different CAC algorithms.

4.4 Measurement based CAC

In real world networks it is increasingly difficult, and virtually impossible, to keep track of all existent and emerging traffic types. In dealing with the large number of connections and a broad spectrum of service types, measurement based CAC algorithms have proven to work very well. A decision whether to accept a new connection is based purely on the current state of the network. The greater the number of connections handled by the CAC, the more optimal this approach is, as the impact of a single connection is close to insignificant compared to the total.

4.5 Parameter based CAC

Parameter based CAC algorithms base the decision whether to admit a new connection on user specified parameters. Such parameters can be peak cell rate, sustainable cell rate, burstiness and other. In its simplest form, a parameter based CAC allocates bandwidth only according to an incoming connection request's peak cell rate. If the sum of all the connections' peak rates exceed the link capacity, the new connection will be rejected. This approach is called non-statistical. The advantage of non-statistical parameter based CAC is that no cells are likely to be lost, as congestions are effectively prevented. The downside is that ATM multiplexing gain is not exploited, thus giving poor bandwidth utilisation.

A more sophisticated parameter based CAC is the statistical algorithm. Making use of the statistical multiplexing gain, this algorithm allocates bandwidth according to a connections expected average rate, which would be somewhere between the peak rate and the sustainable rate. Since the sum of all connections' peak rates may exceed the link capacity, the chances of network congestion are present, although the likelihood may be small.

Successful implementation of parameter based CAC relies on appropriate modelling of all traffic types carried on the network. This is a great disadvantage given the vast variety of different network applications and traffic types. However, due to its simplicity and non-congestion guarantee, non-statistical

CAC is popular where no loss is important.

4.6 Aggregate effective bandwidth estimation CAC

A hybrid between measurement and parameter-based CAC algorithms. This algorithm achieves better bandwidth utilisation than parameter-based algorithm by taking into account the effect of ATM multiplexing gain. This is done by adapting the effective bandwidth calculated using characterisation parameters with measurements of the aggregated effective bandwidth.

4.7 CAC discussion

Naturally, lacking a network simulator module, the dimensioning software tool will not implement any CAC schemes, but basic understanding of such schemes and their application areas is needed. In order to make calculations viable, certain assumptions must be made about network traffic and traffic control mechanisms. One important issue to consider is what type of CAC the network to be planned will implement, so that analysis can be done to estimate likelihood of congestion or percentage of wasted bandwidth.

For any telecom operator a major concern would be to offer adequate GoS to its customers. The need for communication networks grows faster than ever before. The need to keep abreast of the development outweighs the need to plan for a perfectly dimensioned network for today's situation [Vla98]. Flexibility is key, thus, over-dimensioning your network will leave sufficient room for future expansion. Keeping this in mind, assumptions that the network will implement a pessimistic CAC may be made. In the event the future network implements a more optimal CAC, additional bandwidth can be squeezed out of the existing infrastructure. The conclusion to be drawn is that dimensioning network capacity assuming a restrictive CAC algorithm can be justified. The report [WS98] argues that measurement-based CAC are superior to the parameter-based in scenarios where the patterns of traffic sources are hard to predict. Despite this fact, the implementation of the dimensioning tool assumes parameter-based statistical CAC. The reasoning is that assuming any measurement-based CAC will need more sophisticated traffic type modelling and a network simulator, and a parameter-based non-statistical algorithm would be overly restrictive. It is believed that parameter-based statistical CAC will leave room for errors in traffic volume estimations and provide satisfactory accuracy. Ideally bandwidth predictions should be as accurate as possible. How closely the predictions of the tool match the real life results remains to be seen. No statistics are currently available to verify the accuracy of the estimation results.

The conclusions made in the CAC discussion are reflected in the design of database tables are designed and classes are designed, and also the bandwidth calculations. The design chapter will explain this matter in detail.

Chapter 5

Design

The tool comprises a number of modules with different responsibilities. This chapter details the design of these, and how they are put together. Important issues in this chapter include design of MS Access database tables and bandwidth calculations as well as other design matters.

5.1 Requirements and specification

Requirements and specifications are split into two types; non-functional and functional. Functional requirements are those that regard the functionality or behaviour of the software product, such as key algorithms and output results. Other requirements, such as colour of the GUI, development tools restrictions, etc. can be classified as non-functional requirements.

5.1.1 Requirements

Non-functional

The tool shall:

-have a flexible window-based user interface.

-access traffic matrices in MS Excel spreadsheets.

-access network scenarios and parameters stored in an MS Access database format. Network parameters would include:

- Names and hierarchy of nodes
- Network topology representation
- Traffic type definitions

-have a front end developed in Visual Basic to enable further development within Eircom.

Functional

The tool shall:

-model different classes of traffic sources whose parameters will be stored in an MS Access database.

-be able to handle different mixes of multiple types of traffic.

-handle scenarios with a two-level hierarchy of exchanges: Exchanges connected to the ATM network or not.

-be able to manipulate network topologies by changing Access database tables.

-grow traffic matrices for a given base date over a period of time and be able to use the new traffic matrix as input to the planning tool.

-provide a growth parameter for each traffic type that will be taken into account when growing the traffic matrix.

Specifications

Meeting the non-functional requirements:
The tool will:
-be developed using the tools mentioned in the requirements, namely:

- MS Access to store network topologies and parameters such as node names and hierarchy, GoS information, routes between nodes, and traffic source information

- Visual Basic for the front end and user interface.

- MS Excel to store traffic matrices and to save tool outputs.

Meeting the functional requirements:
The tool will:
-model two different traffic classes:

- VBR (Video on demand, speech, etc.)

- CBR (Telephone calls, real time services, etc.)

-produce output in the form of number of ATM bandwidth requirements needed.
-produce output in the form of PSTN bandwidth requirements needed in different network segments.
-compare required ATM bandwidth and required PSTN bandwidth.
-take a base date traffic matrix given in Erlangs as input parameter and grow it over a period of time so that traffic matrices from different points in time can be used as simulator input.
-take the traffic mix as input.
-each exchange in the network will have a traffic mix template associated

with it. The traffic mix will be applied to traffic originating at the node.
-calculate total aggregated traffic mix for links.
-let the user manipulate the network topology by editing MS Access tables using the Visual Basic GUI.
-let the user add new traffic types and classes.
-represent network topology and parameters in a comprehensible way.

5.2 Traffic matrix

The traffic matrix holds information about the network traffic to and from all the nodes in the network. In order for the software tool to recognise the traffic matrix, it must be presented in a specified format. The format has been defined by Eircom profesionals and is similar to the traffic matrix format used for their PSTN planning tool. Below is a sample of a standardised traffic matrix.

Demo.xls

	A	B	C	D	E	F	G
1	MATRIX	To					
2	**From**	**BDTD**	**BLPD**	**BRID**	**CLDD**	**CLTD**	
3	**BDTD**	6636.9	1226.552	580.6396	870.1775	493.3056	
4	**BLPD**	117.596	5756.927	197.0452	467.5476	1381.666	
5	**BRID**	443.2066	192.7519	4772.021	334.033	132.6762	
6	**CLDD**	793.9983	520.8473	293.5173	4139.393	212.2588	
7	**CLTD**	581.2209	1263.141	119.1878	221.7466	1950.31	
8							
9							
10							

Figure 5.1: The traffic matrix format

The traffic matrix must:

- Be stored in an MS Excel 97 (or newer) file format

- Be positioned in the upper left corner of sheet 1
- Start with the word “Matrix” in the upper and leftmost corner. This cell is called the starting coordinate of the matrix
- Contain the word “From” in the first column of the row immediately following the starting coordinate
- Contain the word “To” in the second column and the same row as the starting coordinate
- Have originating exchange ID’s as rows
- Have terminating exchange ID’s as columns The text in the cells can be formatted to suit the users preference. No specific formatting is necessary. No upper limit has been defined for matrix size, but it has to be symmetric, i.e. number of rows is the same as number of columns. This effectively means there is no upper limit to the number of exchanges in the network.

With the layout of the traffic matrix established, a key problem is what to put in it. At the early stages when effort was focused on other, more fundamental issues, it was assumed that traffic would be measured in erlangs, which is standard unit used in PSTN planning traffic matrices. Problems with this approach were later discovered. A key feature of the tool is to enable the analysis of different types of traffic multiplexed together on the network capacity requirements. Thus, it is necessary to express different traffic types with the same units of measurement. After much research in the area of traffic measurements, no easy way to convert between different units has been discovered. For details on this matter, see section 5.4.

Various ideas how to measure traffic were presented and discussed. One suggestion was using two separate traffic matrices, a standard matrix for telephone traffic with units in erlang, and one for data traffic with units in Mbits/s or cells/s. This approach, however, does not solve the problem. Even if the individual matrices represent traffic in a uniform way, the tool still needs to convert between units.

Two obvious solutions are to either convert all traffic into erlangs, or convert all traffic into Mbit/s. See the discussion in section 5.4 for a discussion on difficulties related to unit conversions.

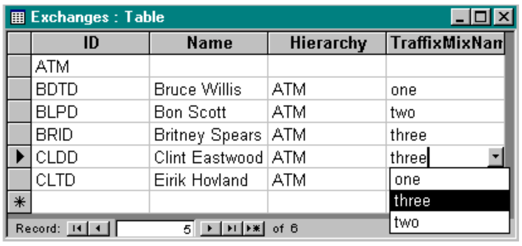
Finally a traffic matrix representing all traffic in Mbits was agreed upon. It was also decided that preparing the traffic matrix was beyond the scope of this project. In other words, all the tool has to worry about is an input traffic matrix with values in Mbits to be sent during the busy hour, and a factor, which estimates the average required bandwidth.

5.3 Access database tables

According to the end users wishes and the tool specification, the network scenarios are stored in MS Access database tables. The tables have been designed with help from Eircom, and are intended to capture all the necessary information about the network topology and traffic needed. Such information includes exchange names, route configurations, traffic type definitions, etc. The following six tables are believed to be a good solution for storing the necessary information.

- Exchanges
- Routes

- TrafficClasses
- TrafficMixNames
- TrafficMix
- TrafficTypes

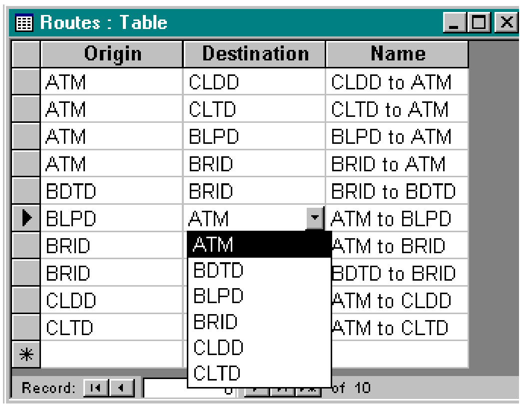

Exchanges : Table

ID	Name	Hierarchy	TraffixMixNam
ATM			
BDTD	Bruce Willis	ATM	one
BLPD	Bon Scott	ATM	two
BRID	Britney Spears	ATM	three
CLDD	Clint Eastwood	ATM	three
CLTD	Eirik Hovland	ATM	

Figure 5.2: Exchanges table

Exchanges table

The “Exchanges” table has four fields. Each change is uniquely identified by its four-letter ID, derived from the exchange’s name or physical location. “ID” is the only field in the table that is required. However, it makes no sense not to specify a value in the “TrafficMixName” field, as the switch will then transmit no originating traffic. The optional name field is where the user can specify a suitable name for the exchange. The name can be chosen by preference, and the value in this field has no effect whatsoever on the calculation made by the tool later on. The third field specifies the exchange’s placement in the network hierarchy. In this application it is assumed that the network is a two-level hierarchy. Either a switch is connected to the ATM connectivity network by a direct physical link, or it is not. It is only the traffic on links between the exchanges with “Hierarchy” set to “ATM” that is of interest; in other words, the traffic on the links that are connected to the ATM connectivity network. The field “TrafficMixName” takes values from the table with the same name. By setting this field, a corresponding traffic mix is associated with the exchange’s originating traffic. Specifying a value for this field is optional, however necessary in order for the exchange to transmit any originating traffic. Traffic passing through the exchange, but not originating at it will not be affected by the exchanges traffic mix setting. An aggregated traffic mix will be computed by adding up all traffic handled by the exchange. The aggregated traffic mix will be calculated independently, and will not be associated with the values entered in the “Exchange” table.

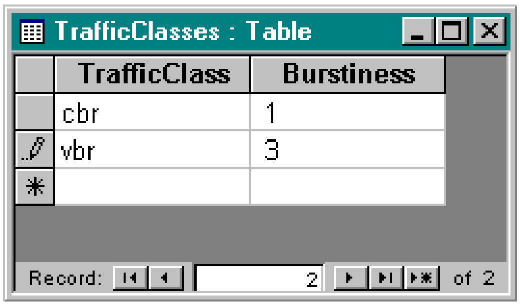


Origin	Destination	Name
ATM	CLDD	CLDD to ATM
ATM	CLTD	CLTD to ATM
ATM	BLPD	BLPD to ATM
ATM	BRID	BRID to ATM
BDTD	BRID	BRID to BDTD
BLPD	ATM	ATM to BLPD
BRID		ATM to BRID
BRID		BDTD to BRID
CLDD		ATM to CLDD
CLTD		ATM to CLTD

Figure 5.3: Routes table

Routes table

This table has three fields. “Origin” and “Destination” are linked to the “ID” field in the exchanges table, and values can be chosen from drop-down lists. Origin and destination refer to the end point of a physical network connection between two nodes. One physical link is in most cases bi-directional, and will be represented with two entries in the table. The rationale for this is to simplify the search for links between two nodes, and also allow for unidirectional links. The "Name" field is holds a descriptive name of the route on the format *[Origin] to [Destination]*. Exchanges appearing in the "Exchanges" table, but not in the "Routes" table will considered to not be a part of the network topology by the program, and hence any traffic to and from these exchanges will be disregarded.


TrafficClasses : Table

TrafficClass	Burstiness
cbr	1
vbr	3

Record: 2 of 2

Figure 5.4: Traffic classes table

TrafficClasses table

Traffic classes can be any of the ATM traffic classes: CBR (Constant Bit-Rate), VBR (Variable Bit-Rate), UBR (Unspecified Bit-Rate) or ABR (Available Bit-Rate). However, currently only two traffic classes have been defined, CBR and VBR. These two guarantee bandwidth and are therefore easier to model than the others. A measure of the level of burstiness is given in the class' level of burstiness is given in the second field. Burstiness is calculated as a traffic class' PCR/SCR (Peak Cell Rate/Sustainable Cell Rate). The "TrafficClasses" table links types of traffic to a more general class of traffic. The number of traffic types is vast, and there is a need for classification. Based on parameters specific to traffic types and a traffic class, many traffic types can be fairly accurately modelled.

TrafficMix table

How to represent a traffic mix in database tables was one of the trickier designs elements of this project. The complexity lies in the traffic mix itself being intuitively a two-dimensional matrix, including traffic type and traffic volume. Keeping a database with multiple traffic mixes adds another dimension to the problem. One solution is to design a database table where one field holds the name of the traffic mix, and another field contains some sort of matrix-like object, for example an excel spreadsheet document. This solution violates the idea of being able to setting up the network scenario by simply editing database tables. Another solution to the problem is to collapse the traffic mix matrix to one dimension. One traffic mix can now be represented as a single record in the table with simple standard type fields. The figure 5.5 shows such a solution.

TrafficMix : Table

TrafficMixName	TelephoneCall	VoIP	Video	FTP	E-mail	New type
Standard	50	10	20	10	10	0
Template one	50	10	10	10	20	0
Template two	60	5	20	5	10	0
Traffic from new nc	50	10	20	10	5	5
						0

Figure 5.5: Traffic mix table old design

This design was initially accepted as the best solution until problems arose when trying to add new traffic types. It was first thought that a small piece of code could work out this problem, but it proved to be more complex than initially imagined. Therefore, a new design for representing traffic mixes was developed. The solution seen in figure 5.6 uses two database tables to represent traffic mixes and is the one used in the final version of the planning tool.

TrafficMix : Table

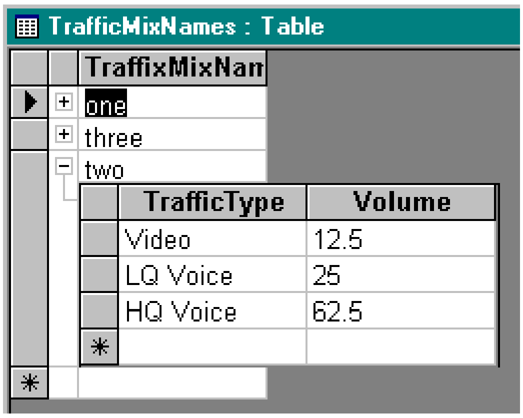
TrafficMixNam	TrafficType	Volume
one	Video	0
one	LQ Voice	0
one	HQ Voice	100
three	Video	0
three	LQ Voice	50
three	HQ Voice	50
two	Video	12.5
two	LQ Voice	25
two	HQ Voice	62.5
*		

Figure 5.6: Traffic mix table

The "TrafficMix" table has three fields. As can be seen from the figure, all traffic mixes are listed in the same table, each with the same number of entries as there are traffic types. The table sorts traffic mixes based on their name, so that the traffic mixes will all appear on consecutive lines. Manipulation of the table is done from the tool GUI. The first column in the table "TrafficMixName", must allow duplicate entries. To simplify searching for traffic mixes, a separate table is provided, functioning as a simple list of all declared traffic mixes. See the following section for details.

TrafficMixNames table

This table serves as some sort of lookup table or a list off all the traffic mixes that have been declared 5.6. The table has only one column, "TrafficMix-Name", which is linked to the field with the same name in the "TrafficMix" table. Opening the sub-table of any of the traffic mixes will display an extract of the "TrafficMix" table representing one single traffic mix.

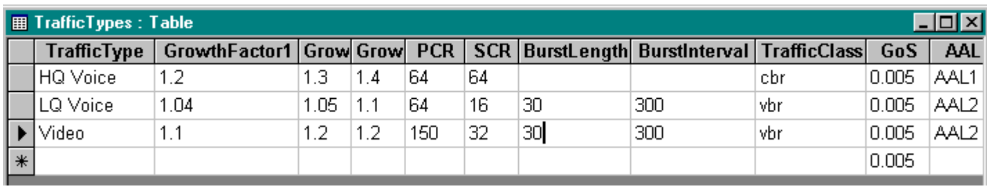


TrafficMixNames : Table

TraffixMixNan
one
three
two

TrafficType	Volume
Video	12.5
LQ Voice	25
HQ Voice	62.5

Figure 5.7: Traffic mix names table



TrafficTypes : Table

TrafficType	GrowthFactor1	Grow	Grow	PCR	SCR	BurstLength	BurstInterval	TrafficClass	GoS	AAL
HQ Voice	1.2	1.3	1.4	64	64			cbr	0.005	AAL1
LQ Voice	1.04	1.05	1.1	64	16	30	300	vbr	0.005	AAL2
Video	1.1	1.2	1.2	150	32	30	300	vbr	0.005	AAL2
									0.005	

Figure 5.8: Traffic types table

TrafficTypes table

The design of the traffic types table has been closely related to the design of the traffic class table. As described in section 4.1.5, traffic types are in ATM classified in five categories. Initially, the traffic class table had most of the parameters that are now in the traffic type table. The intention was to let the traffic class determine the characteristics of the traffic by associated each traffic type with a class. However, there are many more traffic types than there are classes. Giving all traffic classified as, for instance, CBR the same characteristics in terms of PCR, SCR and GoS is hardly desirable. Therefore, most parameters that describe traffic behaviour are now in the traffic type table. "TrafficType" holds the name of the traffic type. "GrowthFactor1" is a factor indicating the expected growth of the traffic type. Traffic can be forecasted for three periods with different growth factors specified in the other growth factor fields. Four fields represent traffic characterisation parameters including maximum bit-rate, sustainable bit-rate, burstLength and burst interval are provided for. The unit for the first two parameters is Mbits/s, and for the last two is milliseconds. The field "TrafficClass" links the traffic type to a traffic class, either CBR or VBR, which determines how effective bandwidth is calculated. End-to-end grade of service requested by the traffic type can be set in "GoS". Finally, the AAL layer used by the respective traffic type can be chosen. Different transmission overhead is added according to the AAL used.

5.4 Code modules

Making an effort to comply with object oriented design principles of this project, the design process set out to identify the different tasks the final program would have to perform. Responsibilities to perform tasks were divided into different modules. Modules developed into fragments of code that fit well together to form cohesive and coherent pieces of code. The following modules were identified:

Routing

The tool needs some way to work out the volume of traffic that traverses a link in the network. Most scenarios used with this tool are expected to have a quite simple network topology. This fact argues for a simple routing algorithm. However, routing should be robust and complex enough to function in even slightly more complicated network topologies. A modified Dijkstra's shortest path algorithm will be used for routing. The details of the algorithm are discussed in section 6.2.2

Database connectivity

Since network scenarios are stored in MS Access database files, the tool needs code to connect to such a database. VB provides some useful features to work with SQL towards databases. Further details on this matter are described in 6.4.

Traffic classes

Traffic classes and traffic types are key issues in this project. The accuracy of the predictions the tool is able to give depends heavily on the traffic models used.

Kruithoff

To provide the functionality to grow traffic matrices from a base date to a planning date, the tool must implement a traffic forecasting model.

The Kruithoff algorithm implemented in this software differs slightly from conventional implementations. Normally, future traffic is predicted by holding the calling rate per customer constant, and then calculate a growth factor according to expected customer base growth. In this implementation of Kruithoff, growth factors are associated with traffic types. The database table "TrafficTypes" has three growth factors, intended for growth over three forecast periods year1, year2, year3. Growth in customer base has not been considered in this project. Any factor related to traffic growth will have to be incorporated into the three growth factors. This effectively means that computation of growth factors is a separate task which lies outside the scope of this project. After calculating actual sums and future sums for the traffic matrix, the normal Kruithoff algorithm is applied to the traffic matrix. For details on Kruithoff refer to section 3.1.3 and 6.2.1 or the document [She80].

The validity of application of the Kruithoff method requires the originating and terminating traffic of all switches in the network to remain constant over the planning period. With traffic growth associated with traffic type rather than customer base growth, the validity can be questioned. With the convergence of data and telephone traffic networks, there is a high possibility this assumption is no longer valid. There is reason to believe that such fundamental changes to the networks offered services will impact the behaviour of the users significantly. Intuitively, a rise in the number of service requests can be expected.

Major changes in infrastructure will also interrupt the normal traffic growth. Changing the core of the network from PSTN to ATM means changing the way traffic is routed and carried over the network, hence invalidating the Kruithoff method.

In times where the network is undergoing radical restructuring like that expected in migrating from single-service networks to NGN, no simple algorithm can be applied to predict the future traffic. The implementation of the Kruithoff algorithm in this planning tool must be seen merely as an additional feature. It may be used to predict traffic matrices over short periods of time, or for analytical purpose.

Unit conversion

The input traffic matrix will be given with units in raw megabytes. In traditional PSTN planning, traffic is measured in Erlangs. One Erlang is one 64Kbit/s circuit occupied for one hour. In PSTN bandwidth is allocated in full circuits. Connections or traffic stream will be given bandwidth corresponding to one or more circuits, hence bandwidth allocation is in multiples of 64Kbis/s. As a unit of measurement the erlang does also not say anything about the number of calls during the hour. One erlang is simply the occupation of one circuit for the duration of one hour. There is a trade-off between number of calls and holding time. Call duration and call frequency have significant impact on the dimensioning of switching equipment, but that issue is outside the scope of this project and will be disregarded. Here it is assumed that one erlang corresponds to one call lasting for an hour. In the ATM world, traffic measurements are often given in cells/s or Mbits/s. A way to compare PSTN traffic and ATM traffic is needed. There has not been much research in this area. Given a set of coarse assumptions about traffic measured in erlangs, it is fairly simple to calculate how much bandwidth the same traffic would consume if it were transmitted over ATM. These assumptions are the following:

- 1 erlang is one call lasting one hour
- 1 erlang occupies one full 64Kbits/s circuit for one hour, regardless of the actual bandwidth needed by the traffic source
- 2 Mbits/s links are provisioned in multiples of 30 erlangs, i.e. 30 circuits. 29 erlangs require one 2Mbits/s link, 31 erlangs require two 2Mbit/s links.
- 64 Kbits/s = 64000 bit/s, not 65536 bits/s

Assuming the traffic on the circuit uses all the available bandwidth, the equivalent of an erlang can be calculated:

```
Total size of an ATM cell = 53 bytes.
Each ATM cell has a header of 5 bytes, and a 48byte payload.
53 bytes = 53*8 = 424bits
64kbs = 64,000bits/sec / 424 = 150.94 ATM cells/s
```

It is also quite easy to calculate the bandwidth requirements for traffic types that do not utilise the 64Kbits/s circuit fully. This is done by calculating the traffic class' effective bandwidth, which in this project is simply the average bandwidth generated by that traffic type. One erlang of traffic type t1, is one connection/source transmitting traffic type t1. The difference between carrying the traffic on PSTN links and ATM links is the savings obtained by getting rid of the unutilised bandwidth in the PSTN circuits. Further discussion on effective bandwidth issues is available in section 4.3.

Traffic mix

Different types of traffic have different characteristics, and will consequently affect the network performance. This project studies the bandwidth requirements for different mixes of traffic in a no-loss case. Clearly the mix of traffic will have the same impact here. The difficulty of accurately predicting the bandwidth requirements depends on the traffic mix. The mix of traffic will be an input to the tool. The user is able to specify the relative volumes of each type of traffic as suits his needs.

GUI

As already mentioned, the GUI is implemented in Visual Basic 6.0. This is in done upon request from the end user to simplify further development. The GUI is window-based and event driven, i.e. responds to user actions as mouse-clicks etc. Effort has been made to separate the GUI from the tool functionality, so that changes can be made as easily as possible.

5.5 Tool output

According to the requirements and specifications, the tool shall produce measurements for required bandwidth in different scenarios. Based on the input scenario setting and the traffic matrix, the tool performs calculations and predicts expected bandwidth requirements on the links between edge switches and the ATM network. Exactly how the calculations are performed is explained in the section 5.4. The tool produces the bandwidth requirement in the event all traffic, data, voice telephone calls or video, were to be carried over the ATM infrastructure. Similar calculations are made to predict the required bandwidth if the same traffic were to be carried over PSTN links.

In the latter case, the calculations rely on the assumption that all traffic types are allocated bandwidth in modularity of 64k circuits. The difference between the two cases is then calculated.

Output from the calculations are written to an MS Excel spreadsheet and may be stored with a user specified file name. The results are also presented automatically on the screen immediately after the necessary calculations have completed. Interpretation analysis of the results are discussed in further detail in section 7.

5.6 Chapter summary

It is difficult to predict the full impact of a fundamental change-over such as the migration from single- to multi-service networks. Many aspects are associated with a high level of uncertainty. Therefore, the design of this tool focuses on flexibility. While capturing the important features, the design leaves room for future extensions and modifications. The implementation of the software reflects this idea by using a modular approach.

Chapter 6

Implementation

The intention of this chapter is to document issues regarding the construction of the code written to realise the software product. The tool used to build this software is Visual Basic 6.0. The software also interfaces with data stored in an MS Access database and MS Excel spreadsheets.

6.1 Classes

All custom designed classes have names on the format clsClassName and each class is saved in a dedicated file with the extension *.cls. The prefix "cls" is chosen in accordance with standard VB naming convention. Classes especially designed for this project are:

- clsStream
- clsPath
- clsLink
- clsTrafficType

- clsTrafficMix
- clsCBR
- clsVBR

clsStream

A stream is a flow of data between two end nodes. Streams are constructed for each of the cells in the traffic matrix, where the total volume of traffic is the value given in the traffic matrix. The stream is therefore the originating traffic of a node in the network going to another terminating node. The stream is not split on its way through the network. Each stream has a traffic mix associated with it. Total traffic is divided between the different traffic types according to the streams traffic mix. In addition to the traffic mix and the total traffic volume, the stream also consists of a path object. The path object is a list of node ID's the stream passes on its way to its destination. Figure 6.1 shows how the different classes are linked.

clsPath

To enable calculating traffic volumes on different network segments, the tool needs a routing algorithm. The routing algorithm is based on Dijkstra's shortest path algorithm and is descried further under routing in section 6.2.2. The routing algorithm builds a list of exchange names from any originating node to any destination node. The path object is then passed to the stream object with the same source destination pair as the path.

clsLink

A link object comprises a number of stream objects which represent all the traffic traversing the link. The link object has the functionality to calculate the sum of traffic carried by the streams. It also computes an aggregated traffic mix by adding up all the traffic of the same type and dividing by the total. It is also the link object that calculates the required bandwidth both for a PSTN scenario and for an ATM scenario.

clsTrafficType

The traffic type object is designed to capture the characteristics of the various traffic types. This class determines how the different traffic types' effective bandwidths are calculated, and other type specific features.

TrafficMix

The functionality of this object is to represent a single traffic mix. In addition to class variables like collections containing traffic types and their relative volumes, the traffic mix implements a function for normalising the traffic mix in the event of the relative volumes not being equal to 100 per cent. Traffic mixes are associated with exchanges. A traffic mix will consist of all traffic types declared in the "TrafficTypes" database table.

clsCBR

Classes have been designed for the two traffic classes that have so far been implemented in the software tool. The CBR class stores class specific parameters.

clsVBR

As the CBR class, the VBR class stores parameters specific to variable bit-rate-type traffic.

6.2 Key algorithms

To a great extent, the code itself contains in-line comments to explain the intention of each module. This chapter provides further detail on the most complex and important algorithms for the functionality of the tool.

6.2.1 Kruithoff

In section 5.4 assumptions regarding the specific implementation of the Kruithoff algorithm used are explained. It is only in preparing the traffic matrix prior to the application of the Kruithoff that this implementation differs from the normal case. The steps described here have not been modified.

For more detail on the Kruihoff algorithm, see section 3.1.3. The algorithm has been implemented as a four step process:

Step 1: For all rows, multiply the all row elements with the originating traffic growth factor; forecast date traffic / sum row elements. The new sum of row elements is now equal to the forecast date traffic.

Step 2: With the new matrix, form the sum of all column elements. If the absolute value of forecast traffic - sum columns exceeds the tolerance, then multiply all column elements in the columns that satisfy the inequality with the terminating traffic correction factor; forecast date traffic / sum column

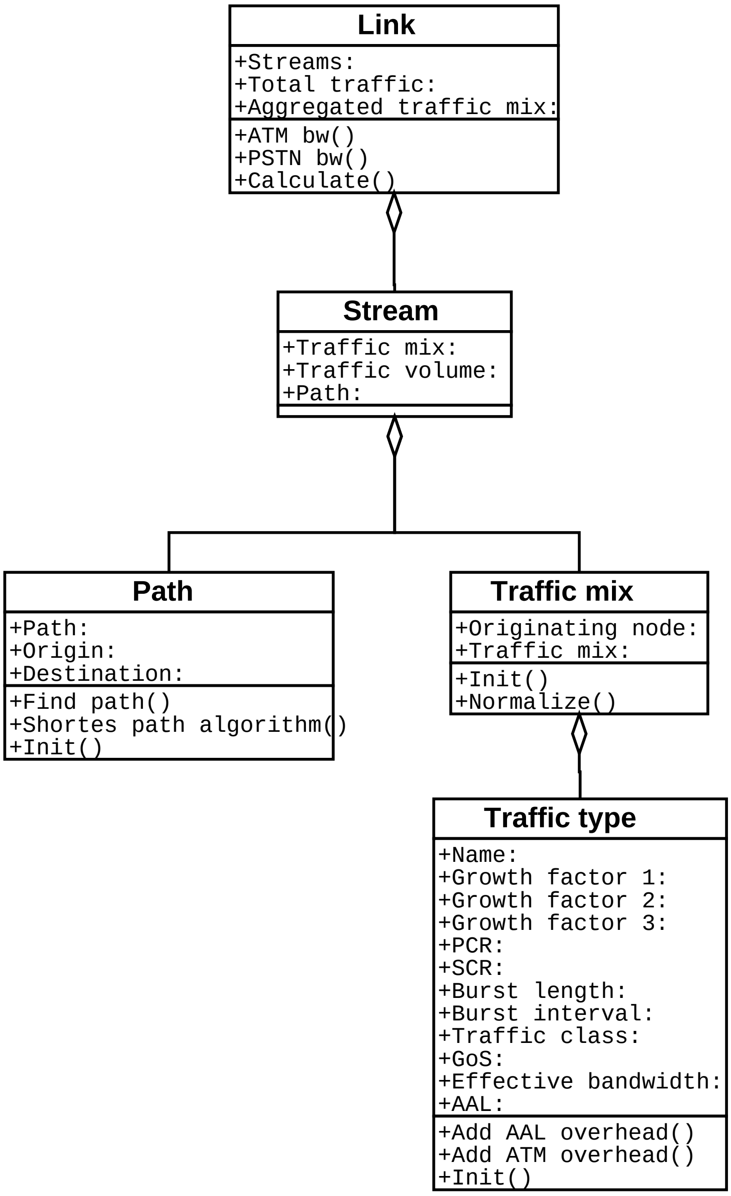


Figure 6.1: Class diagram

elements. The sum of the new column elements should now be equal to the forecast date traffic.

Step 3 With the new matrix, form the sum of all column elements. If the absolute value of forecast traffic - sum rows exceeds the tolerance, then multiply all row elements in the rows that satisfy the inequality with the originating traffic correction factor; forecast date traffic / sum row elements. The sum of the new row elements should now be equal to the forecast date traffic.

Step 4 Repeat step 2 and 3 until sum elements for both rows and columns are within the tolerance.

After this operation has been performed, a unique traffic matrix for the forecast date has been produced.

6.2.2 Routing

The routing mechanism provides the basis for computing aggregated traffic on different network segments. For robustness and flexibility, the algorithm implemented is a version of Dijkstra's shortest path. As the name suggests, this algorithm guarantees to find the shortest path through the network. The shortest path is not necessarily the route with the geographically shortest distance. A principle of weighing routes depending on different factors is often used to compare routes. Sometimes it may be an advantage to find the route with the highest capacity route. In the event of congestion, it may be better to look for a route with the smallest possible delay, small packet loss probability, and so on. A combination of all these parameters can also be used. It is possible to construct quite complex rules for what constitutes the best route. This implementation uses a very simple approach. Distance of the route is measured in number of hops, the number of switching elements passed on the path from A to B. This makes sense since no routes are dimensioned. Route capacity in a no-loss case is what the tool intends to predict, so hop-number is the best option.

The implementation of the routing algorithm is contained in the clsPath class module and the function is called *colSPFAlgorithm*. The algorithm follows these steps:

Step 1: Mark the originating node as permanent and set it as the working node. Set distance to all nodes to infinity.

Step 2: Mark all neighbouring nodes as temporary. Measure the distance to each neighbouring node. If the newly found distance to a node is shorter than found before, mark the node with new distance and from what node the distance was measured.

Step 3: Select the temporary node with the smallest distance. Set this node as the new working node and mark it as permanent.

Step 4: Repeat step 2 and 3 until all nodes have been marked permanent.

Step 5: Find the nodes on the shortest path by backtracking from the destination node.

Once a node has been marked permanent, it is guaranteed that the shortest path from the origin to that node has been found.

6.2.3 Run function

Perhaps the most important function of the tool is the *initializeTool* function in the MainFunctions code module. This function is called when the user clicks on the "Run" command button. The function expects that a traffic matrix has been loaded, and that the network scenario has been properly set up. A few checks are performed before the algorithm starts its main work. First of all, an error message is generated if either the scenario database file or a traffic matrix file is missing. Then, a check is performed to verify that the loaded traffic matrix matches the scenario settings. A typical example of a traffic matrix and scenario mismatch is when routes are set up between exchanges with IDs not present in the traffic matrix. The traffic matrix is also validated to see if its format complies with the specification. Specification of the traffic matrix format can be found in 5.2. After the tool is happy with the traffic matrix and scenario settings, the InitializeTool function starts doing its work. The execution of the function can be describe by the following steps:

Step 1: Perform checks

Step 2: Convert traffic matrix from MB to bit-rate. A function is called that translates the cells in the traffic matrix from megabytes to average bit-rate. It is assumed that the network traffic behaved somewhat uniform in the period when the traffic matrix was recorded.

Step 3: Build paths between each source/destination node pair by invoking the routing algorithm described in section 6.2.2.

Step 4: Build streams for each source/destination node pair by adding the path for that node pair and the traffic between the respective nodes to the stream object. Add stream object to collection of stream objects.

Step 5: Build link objects for all physical links. Physical links are found in the "Routes" database table. A link is built by adding all streams traversing that link to a collection.

Step 6: For each physical link, extract and sum traffic volumes from the streams on the link. Also find the mix of different traffic types for the aggregated traffic.

Step 7: For each link, calculate the ATM bandwidth needed to carry the offered traffic. Also calculate the equivalent bandwidth needed if the same traffic were to be carried over PSTN.

The actual calculation of bandwidth is isolated from the code in this function by function calls, and can easily be modified. This function is the core of the tools functionality.

6.3 Bandwidth calculations

The calculation of bandwidth on the different network links assumes that the traffic is given in megabytes and that the number of connections is somewhat constant during the time the traffic matrix was recorded. Before calculations are performed, the total busy-hour traffic volume is averaged over the hour, giving an average bit-rate. The average rate is distributed among the different types of traffic according to the traffic mix for the respective link. The the number of connections and effective bandwidth of each type of traffic are calculated.

```
EBW = (PCR * MBS + SCR * BurstInt)/(MBS + BurstInt)
```

The next step is then to calculate link capacity requirements. For ATM this is done by adding the ATM transmission overhead to the effective bandwidth for each of the connections.

```
ATM BW = EBW + ATM overhead
```

For the PSTN scenario, circuits are allocated according to the peak cell-rate. For VBR traffic, there will be a gap between the PCR and the effective bandwidth of the connection. This gap represents the amount of wasted bandwidth.

```
numCircuits = PCR/64
If PCR mod 64 = 0 then
PSTN BW = numCircuits * 64
Else numCircuits = (Integer part of numCircuits + 1) *64
```

The bandwidth saving obtained is the difference between the PSTN allocated and the ATM bandwidth utilised.

```
Bandwidth saving = PSTN BW - ATM BW
```

6.4 Implementation issues

Contrary to popular belief among object oriented programmers, VB does have object oriented development facilities. The language is particularly powerful when it comes to working with other Microsoft applications. Visual Basic provides an easy and efficient way to create objects for handling SQL database operations, and also objects for interfacing with data in Excel format. The enabler of these features is Microsoft's OLE services.

When implementing the software, effort has been made to adhere to the principles of object oriented programming. The code has been organised as far as possible into cohesive and coherent modules. A key concern has been to properly document every part of the code with in-line comments to simplify further development.

For maximum flexibility, code for presentation has been separated from code responsible for functionality. As an example, code contained in form modules will be kept at a minimum for handling the GUI. Whenever "real work" is needed, function calls will be made to code located in other modules.

Most database operations performed by the program are read-operations. Yet, for data integrity, database operations are encapsuled as transactions where there is a possibility of conflict.

At an early stage of the project, it was considered to write some modules in languages other than Visual Basic. This option might have proved to be a good idea, but was left due to time constraints. However, it might be an idea for further development. Especially if it could help to boost the speed of heavy calculation processes.

The software was constructed using a spiral model for implementation. Progressively, bits and pieces of code implemented for handling small tasks, have been built together to form the end product. Since design has undergone modifications a number of times from when the project was started, this has proved useful.

6.5 MS Access version compatibility

For the convenience of further development MS Access 97 format is preferred. However, since this version of the software was not available to the developer, MS Access 2000 was used in design of the database tables. Before distribution, the database was converted to an Access 97 version. In doing so, there was a concern that errors might occur, since the VB code that interfaces to the database files was written for an Access 2000 version. Care was taken in the implementation process to avoid compatibility problems. After converting the database files to a 97 version, extensive tests were carried out, concluding there is full compatibility.

6.6 Installation

Installation of the software should be quite simple. System requirements are:

- Microsoft 98 platform or later.
- MS Access 97 or later
- MS Excel 97 or later

The following list describes all the steps necessary:

1. Copy source code a directory of preference.
2. Modify constants under the section "Files and directories" in the file "PublicConsVars.bas" to match the location of the source code.

3. Make sure the database file “Matrices.mdb” is stored in the directory “Template” under the location of the source code.

4. ODBC Data Sources (32bit) registry a driver for the “Matrices.mdb” file. Consult windows help files for assistance

5. Make sure the MSHFlxGd.ocx file is installed in the users Microsoft Windows System directory. The file should already be installed by default.

6.7 Chapter summary

The implementation chapter explains important decisions taken regarding challenges encountered during the development of the software. Construction of classes and code modules alon with key algorithms have been described. This chapter also provides a step-by-step guide to installation and running of the software tool. The tool outputs and evaluation of the results are discussed in chapter 7.

Chapter 7

Results

The main output of the planning tool are the results of the bandwidth calculations. This chapter evaluates the results computed from a set of different test scenarios. Test scenarios have been constructed to show the effect of different traffic mixes.

7.1 Test configuration

Testing to validate the functionality of the tool has been performed. Programming bugs exposed during this process have largely been removed. For collection of results, the tool was fed a five-node traffic matrix with values taken from a real Eircom traffic matrix. The base matrix had units in erlang. Before using the matrix for calculations, the erlang values were converted into megabytes. Although the tool is capable of handling larger network scenarios, a five-node scenario matching the traffic matrix was constructed. Theoretically, there is no upper limit to the size of the traffic matrix. In practice, it depends on the memory capacity of the environment the program is run. The diagram 7.1 below shows the traffic matrix used in these tests.

Demo.xls

	A	B	C	D	E	F	G
1	MATRIX	To					
2	**From**	**BDTD**	**BLPD**	**BRID**	**CLDD**	**CLTD**	
3	**BDTD**	6636.9	1226.552	580.6396	870.1775	493.3056	
4	**BLPD**	117.596	5756.927	197.0452	467.5476	1381.666	
5	**BRID**	443.2066	192.7519	4772.021	334.033	132.6762	
6	**CLDD**	793.9983	520.8473	293.5173	4139.393	212.2588	
7	**CLTD**	581.2209	1263.141	119.1878	221.7466	1950.31	
8							
9							
10							

Figure 7.1: Traffic matrix

Routes were established to connect exchanges to the ATM connectivity network in a star topology. Traffic types were set up according to the database table in the diagram 5.8.

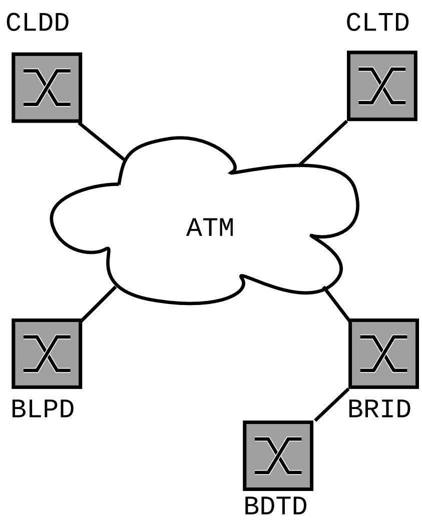


Figure 7.2: Test network

The first test predicts the impact of only one traffic type; pure voice traffic transmitted using AAL1 circuit emulation. The calculations made by the tool produced the results in the figure 7.3.

As can be seen, the results show a rise in bandwidth requirements. Carry-

Results.xls

	A	B	C	D	E	F	G	H	I	J
1	Scenario name:		C:\My Documents\Dissertation Project\Development\Planning Tool\Demo.mdb							
2	Traffic matrix name:		C:\MY DOCUMENTS\DISSERTATION PROJECT\DEVELOPMENT\PLANNING TOOL\Demo.xls							
3										
4										
5	Link name		% HQ Voice	% LQ Voice	% Video	Avg. Bitrate	PSTN BW	ATM BW	Difference	Diff %
6	CLTD to ATM		100	0	0	9.78933978	9.792	11.7917	1.999705	20.42182
7	ATM to BRID		100	0	0	11.8934704	11.904	14.32623	2.422226	20.348
8	CLDD to ATM		100	0	0	8.25361422	8.256	9.941853	1.685853	20.41974
9	BRID to BDTD		100	0	0	11.3482144	11.328	13.66944	2.34144	20.66949
10	ATM to BLPD		100	0	0	11.9269933	11.904	14.36661	2.462606	20.68721
11										
12	Totals					53.2116322	53.184	64.09583	10.91183	20.51713
13										

Sheet1 / Sheet2 / Sheet3

Figure 7.3: CBR only

ing pure 64kb voice over ATM does not appear to be very economical. This may come as a surprise, but the solution is simple. The extra bandwidth requirements are due to the ATM and AAL overhead. It is mainly for VBR traffic that ATM bandwidth savings start to take effect.

The second test scenario uses the same network topology as the first one. This time, a new traffic type is introduced; compressed voice. The traffic mix is 50 per cent of each type.

Results.xls

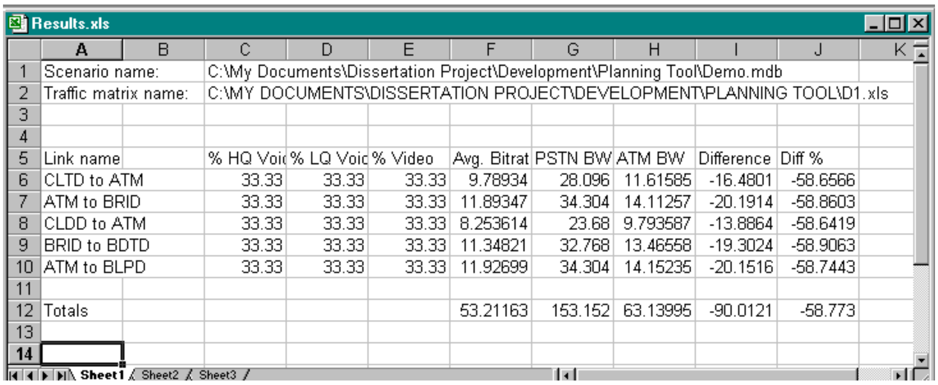
	A	B	C	D	E	F	G	H	I	J	K
1	Scenario name:		C:\My Documents\Dissertation Project\Development\Planning Tool\Demo.mdb								
2	Traffic matrix name:		C:\MY DOCUMENTS\DISSERTATION PROJECT\DEVELOPMENT\PLANNING TOOL\Demo.xls								
3											
4											
5	Link name		% HQ Voic	% LQ Voic	% Video	Avg. Bitrat	PSTN BW	ATM BW	Difference	Diff %	
6	CLTD to ATM		50	50	0	9.78934	20.224	11.66069	-8.56331	-42.3423	
7	ATM to BRID		50	50	0	11.89347	24.64	14.16705	-10.473	-42.5039	
8	CLDD to ATM		50	50	0	8.253614	17.088	9.831388	-7.25661	-42.4661	
9	BRID to BDTD		50	50	0	11.34821	23.552	13.51756	-10.0344	-42.6055	
10	ATM to BLPD		50	50	0	11.92699	24.704	14.20698	-10.497	-42.4912	
11											
12	Totals					53.21163	110.208	63.38365	-46.8243	-42.4872	
13											

Sheet1 / Sheet2 / Sheet3

Figure 7.4: CBR and VBR voice

The results from the second test in figure 7.4 show the ATM statistical multiplexing taking effect. Bandwidth requirements are now smaller then would be the case if the same traffic were to be carried over a PSTN link.

The third test adds another new type of traffic source. Video streaming is introduced onto the network. The traffic mix distributes the total traffic evenly between the three types. According to the tool output seen in figure 7.5, drastic bandwidth savings are obtained. The percentage of saved bandwidth is expected to be affected both by the total volume and the traffic parameters of the video traffic. If a source requires 1.1 circuit, two PSTN circuits will be allocated so the relative savings will be large by going moving to ATM. If the source requires 8.1 circuits the saving will be relatively smaller.


Results.xls

	A	B	C	D	E	F	G	H	I	J	K
1	Scenario name:		C:\My Documents\Dissertation Project\Development\Planning Tool\Demo.mdb								
2	Traffic matrix name:		C:\MY DOCUMENTS\DISSERTATION PROJECT\DEVELOPMENT\PLANNING TOOL\D1.xls								
3											
4											
5	Link name		% HQ Voic	% LQ Voic	% Video	Avg. Bitrat	PSTN BW	ATM BW	Difference	Diff %	
6	CLTD to ATM		33.33	33.33	33.33	9.78934	28.096	11.61585	-16.4801	-58.6566	
7	ATM to BRID		33.33	33.33	33.33	11.89347	34.304	14.11257	-20.1914	-58.8603	
8	CLDD to ATM		33.33	33.33	33.33	8.253614	23.68	9.793587	-13.8864	-58.6419	
9	BRID to BDTD		33.33	33.33	33.33	11.34821	32.768	13.46558	-19.3024	-58.9063	
10	ATM to BLPD		33.33	33.33	33.33	11.92699	34.304	14.15235	-20.1516	-58.7443	
11											
12	Totals					53.21163	153.152	63.13995	-90.0121	-58.773	
13											
14											

Sheet1 Sheet2 Sheet3

Figure 7.5: CBR and VBR voice and VBR video

7.2 Analysis of results

Analysing results produced by the tool, one should keep a few things in mind. The network scenarios the tool works with, do not exist yet. Hence, there are no statistical data from real-life network available for comparison and validation of the calculations. Evaluation must at the current state be based on network planning expert opinion. Another thing to remember is that coarse assumptions have been made about traffic sources. Traffic source modelling is based on parameters specified by the user of the tool. Although a flexible solution, the issue of accuracy must be seen in close relation to the assumptions about CAC algorithm made in chapter 4.3. Calculations also rely on the correctness of the input traffic measurements. It is difficult to predict the error in future traffic estimations since the network infrastructure is undergoing fundamental changes in the migration from PSTN to NGN.

Chapter 8

Conclusion

8.1 Future work

The main output of this project is a software tool for NGN capacity dimensioning. Some assumptions have been made in the absence of any previous work in this particular area. For the future, it will be necessary to validate the calculations made by this software tool. One way to achieve this is to simulate the scenarios processed by the tool. In particular, it would be interesting to see how closely the traffic types in this project match the behaviour of traffic in a real network.

In the next version of the software, GoS predictions should be included. Due to time constraints, this feature has not been implemented in this version. By using the proposed multiplexing gain factor to dimension ATM links, and then a random number generator to compute the probability of all traffic sources simultaneously transmitting at peak rate, GoS functionality can easily be added.

During migration phases, different physical networks will have to interface with each other. In the breaking point between the core NGN and the edge nodes, different transport mechanisms and media meet. Different physical layers introduce different traffic overhead. It would be interesting to investigate the impact of a uniform physical layer versus scenarios with heterogenous physical layers. There is reason to believe further cost savings can be achieved.

So far, only bandwidth requirements at the edge of the network are calculated. This is only the first step. Future enhancements can add functionality to dimension the core of the network.

Visual Basic is not designed for speed. Heavy matrix operations performed by the tool could be sped up using other development tools. Database operations also add to the computational cost. Modification could be made to boost the performance of the tool, even is speed is not the most important issue in this application.

8.2 Obstacles overcome

Initially the plan was to design a simple network simulator to simulate the traffic generated by the various traffic input sources. Much effort was spent studying queuing theory and ATM traffic behaviour on the cell level. The studies of the NIST ATM network simulator provided a framework for what was supposed to become a simple discrete event cell level network simulator incorporated in the planning tool. The rationale for designing a small network simulator was to enable the calculation of link capacity on the different network segments. The [ea98b] provides in addition to operation manual models for traffic source models used in the simulator.After furhter research it was believed that a continuous time simulator model would be a better option.

Later yet, the idea of implementing a network simulator was abandoned altogether, when talks with experts revealed that the dimensioning was supposed to be conducted on a higher level. Instead of worrying about loss of individual ATM cells, the network capacity dimensioning will be carried out for scenarios with a vast number of connections, thus eliminating the need for accurate prediction of cell loss. From a planning and business point of view it is more important to make sure enough bandwidth is available to carry the expected traffic load, and thereby securing revenue.

The output of the developed software suggests, as expected, that there is substantial economical benefits to be obtained by moving to a multi-service network infrastructure. Although difficult to evaluate the accuracy of the tool's calculations, the results allow quantification of bandwidth savings, and hence, gives an indication of cost savings. As discussed in section 7, assumptions have been made. The validity of these assumptions may be questioned, but the results will still be valuable.

A major challenge has been to find a proper way to represent traffic in an environment that merges standard PSTN voice, multimedia and data traffic. The source of the problem being different units of measurements used for different types of traffic. In traditional PSTN planning, traffic is measured in erlang, which is the average number of calls in the network at any instant during the busy hour. In data communication, traffic can be measured as average bit-rate.

At some stage, it was proposed to use two different traffic matrices. One representing the data traffic, and one for telephone traffic. This approach was later abandoned. In the end all traffic must be translated into the same unit of measurement later anyway. The only advantage with having two separate traffic matrices would be that pre-processing of the matrix would not be necessary. However, relieving the user of the problem of pre-processing the traffic matrices, the complexity of the software implementation is increased.

Working with a single traffic matrix, the responsibility to compute the matrix lies with the user. Nevertheless, the tool needs knowledge about the format of the input traffic. Integrating voice and data, the question of how to represent the bursty VBR traffic arises. A matrix with units in average generated bit-rate is not a good option. This approach relies on knowledge of the different types of traffic and the mix of traffic in advance. Another problem with this traffic matrix format is whether the traffic is from a PSTN or ATM network.

An approach similar to PSTN planning can be adopted. Erlang is basically a measure for the volume of traffic offered to the network during the busy hour, and can easily be converted into megabytes. ¡¡referance to equations¿¿ It is also possible to measure the sheer volume of data traffic during the busy hour, and then compile a traffic matrix with units in megabytes. Using this approach, assumptions must be made that the traffic behaviour is somewhat stationary over the busy hour, or over the interval when traffic was recorded, with the implication that the number of traffic sources generating traffic is close to constant. This is the traffic matrix format that was finally agreed upon. It is flexible in that it does not require any a-priori knowledge about traffic types.

The term busy-hour is used to describe the time of the day, week and month when the heaviest load on the network. An interesting problem is to identify the busy-hour for a network with both data traffic and telephone traffic. There is reason to believe that the intensity of telephone traffic and data traffic does not peak at the same time of the day. An office worker may spend half an hour in the morning checking his e-mail and reading the news on electronic newspapers, and then maybe make most of his telephone calls during the afternoon. It has not been investigated how this issue affects the computing of the traffic matrix. However, the tool leaves the user with the opportunity to experiment with different situations.

8.3 Project's contribution

In times of change for the telecom industry, there is a need to develop new methods for network planning and dimensioning. This project has created a software tool to assist network planners in a migration phase from PSTN to NGN. The outputs of the tool will contribute to generating new knowledge in an area where experience is lacking. Given the uncertainty of issues regarding the future situations and the impact of integration network integration, the tool provides flexibility in analysis of different network scenarios, and is a solid framework for further development. The results of this work can serve as an argument for justifying costly investments in new technology.

Bibliography

[Don00] M. Donohoe. Eircom next generation network. Plan report 1, Eircom, 2000.

[dP] M. de Prycker. *Asynchronous Transfer Mode*, chapter 3, pages 100–121. ISBN 0-13-053513-3.

[ea98a] A. O'Neill et. al. An overview of internet protocols. *BT Technol J*, 16(1):126–139, 1998.

[ea98b] N. Golmie et. al. The nist atm/hfc network simulator, operation and programming guide, version 4. Technical report, NIST, 1998.

[HH99] C. Steinkellner H. Hlavacs, G. Kotsis. Traffic source modeling. Review, University of Vienna, 1999.

[JVG99] B. Vanneuville J. Van Gorp. Voice over ip and the next generation network, response to the art consultation on internet telephony. Technical report, Level 3 International, 1999.

[MHl00] G. Öster M. Hallenstå l, U. Thune. Engine server network. Technical report, Ericsson, 2000.

[MRI99] D. S. Reeves M. R. Izquierdo. A survey of statistical sources models for variable-bit-rate compressed video. *Multimedia Systems*, (7):199–213, 1999.

[Ryu96] B. K. Ryu. The importance of long-range dependence of vbr video traffic in atm traffic engineering: Myths and realities. Technical report, Columbia University, NY, US, AT&T Bell Laborarories, 1996.

[She80] M. J. Sheridan. Traffic forecasting in telecommunication networks. Technical report, PT, 1980.

[Swe01] T. Sweeney. Next generation networks: The future of business. *NewsLink*, 9(2):13–17, 2001.

[Tek01] M. Tekel. One network for voice, data and multimedia services. Technical report, Alcatel, 2001.

[Tel98] Ericsson Telecom. *Understanding telecommunications*, volume 2. Ericsson Telecom, Telia and Studentlitteratur 1998, 1 edition, 1998.

[Tia99] Y. Tian. A survey on connection admission control in atm networks. Survey, DePaul U. Chicago, US, 1999.

[Vla98] S. Vlaar. Atm structure otimisation design and dimensioning in a fast moving market. Technical report, KPN Research, 1998.

[WS98] D. Liu W. Sun. Call admission control: A survey for effective bandwidth-based approach and measurement-based approach. Survey, Stevens Institute of Technology, Hoboken, NJ, US, 1998.