VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND TELECOMMUNICATION TECHNOLOGIES

DSL-BASED TRIPLE-PLAY SERVICES

BACHELOR THESIS

AUTHOR **MHANNAD AL-ANQARI**

SUPERVISOR **ING. PAVEL ŠILHAVÝ, Ph.D.**

BRNO 2009

Poděkovaní

Děkuji vedoucímu bakalářské práce Ing. Pavlu Šilhavému, Ph.D., za velmi užitečnou metodickou pomoc a cenné rady při zpracování bakalářské práce.

V Brně dne ……

(podpis autora)

TABLE OF CONTENTS

Introduction **1**

1. DSL..... **2**
1.1 High Data-Rate Digital Subscriber Line (HDSL)..... 3
1.2 Integrated Digital Subscriber Line (IDSL)..... 3
1.3 Single-Line Digital Subscriber Line (SDSL)..... 3
1.4 Very High Speed Digital Subscriber Line (VDSL)..... 4
1.5 Asymmetric Digital Subscriber Line (ADSL)..... 4
1.5.1 CAP and DMT..... 4
1.5.1.1 Carrierless Amplitude and Phase (CAP)..... 5
1.5.1.2Discrete Multi-tone (DMT)..... 6

2. ATM and IP **8**
2.1 ATM 9
2.1.1 Concepts and Background..... 9
2.1.1.1 ATM Cell Basic Format..... 10
2.1.1.2 ATM Cell Header Format..... 11
2.1.1.3ATM Cell Header Fields..... 11
2.1.2.1 ATM Network Interfaces..... 12
2.1.2.2 ATM Devices..... 13
2.1.2.3 ATM Services..... 13
2.1.2.3.1 ATM Virtual Connections..... 14
2.1.2.3.2 ATM Switching Operations..... 14
2.1.3.1 ATM Reference Model..... 15
2.1.3.1.1The ATM Physical Layer..... 16
2.2.4.1.2 ATM Adaptation..... 17
2.3.1 ATM Connections..... 19
2.3.1.1 ATM and Multicasting..... 20
2.3.1.2 ATM Quality of Service..... 21
2.3.1.3 Data Encapsulations..... 21
2.3.1.3.1 LAN Emulation..... 22
2.3.1.3.2 Multiprotocol over ATM..... 23
2.3.1.3.3 RFC-1483..... 24

3. Triple play..... **25**
3.1 INTERNET PROTOCOL (IP)..... 26
3.1.2 ROUTERS AND HOSTS..... 26
3.1.3 IP ADDRESSING..... 27

3.1.4 IP HEADER…… 27
3.2 VOIP……... 31
3.2.1 Issues in a VoIP Network……. 32
3.2.1 Service set……..….. 33
3.2.2 Choice of Signaling Protocols……..... 33
3.2.2.1 The Session ignition Protocol …… 34
3.2.3 Quality of Service…….. 34
3.2.4 Reliability / Availability……. 37
3.2.5 Call Routing and Number Plans……...….. 37
3.2.6 DTMF and Other Tones and Telephony Events…….. 38
3.2.7 Firewall and NAT traversal…….. 38
3.3 IPTV…… 39
3.3.1 Video Doubles Communications ARPU…… 39
3.3.2 Transmission……. 40
3.3.3 Codecs…….. 41
3.3.4 IPTV Architecture…….. 43

4.Monitoring softwares…… 44
4.1 Wireshark……..…… 44
4.1.1 The "Packet List" pane…… 45
4.1.2 Capturing Live Network Data……..… 46
4.1.3 The "Capture Interfaces" dialog box……..… 46
4.1.2.1 While a Capture is running..... 47
4.1.2. 2 Finding packets……..… 48
4.1.2.3 Filtering packets while viewing……... 48
4.2 OmniPeek…… 50
4.2.1 Voice and Video view window…… 50
4.2.1.1 Voice and Video upper pane views …… 51
4.2.1.1.1 Calls view …… 51
4.2.1.1.2 Media view ……. 52
4.2.1.2.1 Voice and Video Details tab …… 53
4.2.1.2.2 Voice and Video Event Summary tab…….. 53
4.2.1.2.3 Voice and Video Event Log tab …….. 54
4.2.2 Voice and Video Visual Expert …… 55
4.2.2.1 RTP/RTCP Rows …….. 56
4.2.3 Making a voice or video filter …… 57

5. Configuration and Measurement …….. 59
5.1 Configuration …… 59
5.2 Laboratory Results…….. 65
5.2.1.1 Public connection without load (high bit rate)…….. 65
5.2.1.2 Public connection with load (high bit rate)…… 67
5.2.2.1 VLAN Connection without load (high bit rate) …… 69

5.2.2.2 VLAN Connection with load (high bit rate) …… 72
5.2.3.1 VLAN Connection without load (Low bit rate) …….. 74
5.2.3.2 VLAN Connection with load (Low bit rate) …… 76
5.2.4.1 Public connection without load (Low bit rate) …….. 78
5.2.4.2 Public connection with load (Low bit rate) …… 81
5.2.5 VLAN Connection with and without load (various bit rate)…… 84
5.2.6 Public connection from far distance (High bit rate)……. 84

6.Conclusion…… 86

Glossary…… 88

References…… 92

Introduction

The DSL is a technological mean of communication which consists of a high bandwidth which branches many formations such ADSL, HDSL, ISDL and many more via communication through a telephone line.

In the history Telecommunications means were limited yet the most frequently used mean was voice communication through the POTS. It commenced its communication through analogue communication which is dependent upon the change of the analog wave, since it was reliant upon human influence. Human influence was essential to establish communication between users.

As time passed, the human involvement became more expensive therefore it had to be substituted with electronic/digital form which identifies the subscriber through a serial number, this serial number contains several signs, as for the change in frequency or pulse to determine the identity of the subscriber. This progress contributed in the development of multiple subscribers which helped in cost saving somehow since remote communication because may be expensive since it goes through several switches,

But that wasn't the end of the ultimate development it was noted that the sound may be transmitted via the Internet, which makes communication less expensive and run the voice over Internet Protocol specifies the users within the network ie for each user there is an IP address. What is known about Internet Protocol is that through it computerized data is linked to a computer network protocols which operates in a group of Internet Protocols.

After the virtue of these possibilities of this feature came up the idea of Internet Protocol Television, since the same concept used for the Voice over Internet Protocol, since there is no need to use Satellite Service for whoever wishes not to use it since it goes through a lot of television channels and high resolution. We have noticed that the user may dispense three entrances, therefore why is there a need for a telephone cable, internet cable and television cable.

We use the internet on a daily basis we can watch the video or television, or a voice call via the Internet but why don't we use one mean to transfer all this data? This concept came up to introduce triple play through one cable which integrates of all data flow through an ADSL modem that is alleged by the intermediary to receive information and signals to the distribution of multimedia and IP DSLAM which alters the packet IP cells of ATM. We will explain these things, especially the mechanism of the formation.

It is well known that ADSL technology enables a fast data transmission up to around 8 Mbps. this high amount provides a possibility of combining several data packages, for example, voice communication with the data.

There are at least four methods which the ADSL deal with, and these are: END-TO-END VCCs, Bridging, PPP, and routing, and each one of these ways require configuration and choosing the correct protocol in order to achieve the communication goal.

In the Laboratory, We will choose the Bridging mode for the communication, which means that the ADSL modem will set as a wire for the data transmission, and this required creating the PVCs which is transmitted through the DSLAM to ADSL modem and then to other devices; pc, IP phone, and set-top box.

We shall proceed with four ports of the ADSL modem as following: The first port will be used for the public connection without the concept of the triple play connection. The second one will be used for the VLAN to the VOIP with the concept of the triple play principle to analyze the voice and integrate it with internet data. As for the third one will be connected to the VLAN for the IPTV again with the concept of the triple play principle. Unfortunately we won't be analyzing the video in our laboratory.

And this is the procedure which I will perform my experiment by which is based on the establishment of a single channel way for data transmission.

1. DSL

Digital subscriber line (DSL) service is a broadband digital transmission service which is offered by normal telephone lines. DSL is usually used by telephone companies to provide data, video, and voice services over these existing copper telephone lines. DSL does not substitute the function and quality of the existing analog telephone service offered on telephone lines, so that both analog phone and broadband services can be simultaneously used.

Prior to the DSLs, latent capacity within the copper cable plant was made obsolete, as traditional voice traffic occupied only 0 to about 4 kHz, less than one percent of the available frequency spectrum. In cities and countries where ISDN is a factor, the copper loop is utilized up to 80 or 120 kHz, a10 percent utilization. Still, the remaining spectrum up to about 1 MHz was left unused due to shortage of sufficiently sophisticated encoding schemes.

Further ahead in this chapter, we shall introduce various types of DSLs. After the introduction to the types we aim to focus specifically on the ADSL technology in

more detail as well as encoding CAP and DMT. CAP was used in the past and now it has been replaced with DMT encoding.

1.1 High Data-Rate Digital Subscriber Line (HDSL)

The first technology which was a kick off in the spectrum was the high-speed digital subscriber line (HDSL), capable of carrying a T1 or E1 (1.5 or 2 Mbps) worth of traffic symmetrically over 2 copper pairs. A more modified form of the technology, known as HDSL2, requires only a single twisted pair. HDSL has seen accepted by the incumbent service providers for delivery of leased line commercial services at T1 or E2 rates, or as a means of carrying multiple voice channels (24 or 30) between a serving CO and a remote terminal. This was proved worthy of its usage, especially in the excessive use of voice capacity for businesses or residential areas. HDSL is still a fairly expensive technology when compared to ADSL, and until HDSL2, most vendor implementations were proprietary (meaning that hardware from a single vendor was required at both ends of the loop).

1.2 Integrated Digital Subscriber Line (IDSL)

The next DSL, the Integrated Digital Subscriber Line (IDSL), has effectively reused ISDN 281Q encoding but for permanent connectivity. Since it dispenses with ISDN's 16 kbps signaling channel (the D channel), its maximum data rate is 144 kbps symmetric over 1 copper pair. This bandwidth is appropriate for telecommuter's usage, where IDSL is primarily marketed.

IDSL has been supported by the CLECs, and to a low extent, the ILECs. A reason for acceptance by the CLECs is to be able to use 281Q encoding. In most locations, it is easier for a CLEC to gain permission to run this encoding over an ILEC's local loops than is the case with alternatives (such as that used by ADSL). Thus the pain in commissioning a service is significantly less with IDSL than with ADSL.

1.3 Single-Line Digital Subscriber Line (SDSL)

The Symmetric Digital Subscriber Line (SDSL)is a single line version of HDSL transfers up to 768 kbps, also over a single twisted pair and is enable to operate over a plain old telephone *service* (POTS) so that a single line can support POTS and T-1/E-1 at the same time.

As with IDSL, it uses 281Q encoding, making it appealing to the CLECs. SDSL has been a mean of entry for CLECs hoping to take leased line business, and in fact has seen successful in this space. It is cheaper to deploy than HDSL and, as noted, requires a single twisted pair unlike most HDSL variants that require two. None of these technologies, HDSL, IDSN, and SDSL, are capable of supporting traditional analog or digital (ISDN) voice traffic in the baseband.

1.4 Very High Speed Digital Subscriber Line (VDSL)

The last of the DSL technologies, the Very High Speed Digital Subscriber Line (VDSL) is just a start off to see deployment in any quantity. Due to a distance restriction of approximately 3000 feet (1000 meters), it is suited to the DLC/FSAN environment as opposed to central office deployment. However, with this distance tradeoff it is brought to a maximum data rate of up to 52 Mbps. Therefore, unlike the other DSLs, VDSL may be used to transfer one or more channels of high-quality video.

1.5 Asymmetric Digital Subscriber Line (ADSL)

ADSL is intended to establish the mean of communication to the customer's premises. It transmits two separate data streams with much more bandwidth devoted to the downstream leg to the customer than returning. It is effective due to symmetrical signals in many pairs within a cable (as occurs in cables coming out of the central office) significantly restricts the data rate and estimated length of the line.

ADSL succeeds if it takes advantage of the fact that most of its target applications (video-on-demand, home shopping, Internet access, remote LAN access, multimedia, and PC services) function perfectly well with a merely low upstream data rate. MPEG movies require 1.5 or 3.0 Mbps downstream but requires only in limits of 16 kbps to 64 kbps upstream. The protocols controlling Internet or LAN access require somewhat higher upstream rates but in several situations we can get by with a 10 to 1 ratio of downstream to upstream bandwidth.

1.5.1 CAP and DMT

The two technologies, CAP and DMT are very different in the process they encode the data across the local loop in. Although both are frequency domain techniques, CAP relies more deeply on the time domain than the DMT does, sending high bandwidth symbols across a wider spectrum for a short period of time. DMT's carriers are just over 4khz, capable of sustaining a bandwidth of about 32 kbps, near to the analog modem (which would be expected, since analog modems operate in the POTS frequency band which tops out at about 4 kHz).However, both fulfill their aims in transmitting data.

The following sections will cover in more depth the technical underpinnings of CAP and DMT in terms of modulation schemes, processing, performance, and interference.

1.5.1.1 Carrierless Amplitude and Phase (CAP)

Carrierless Amplitude and Phase (CAP) is a DSL encoding technique which relies upon single downstream and upstream carriers occupying a larger segment of the available bandwidth. Figure **1.1** depicts this spectrum, including the POTS traffic in the baseband. The technology segments the available spectrum into two single carriers, with the upstream between f1 and f2 and the downstream between f3 and f4. The figure also displays the Power Spectral Density (PSD) of the signals, with the upstream at -38dBm/Hz and the downstream at-40.

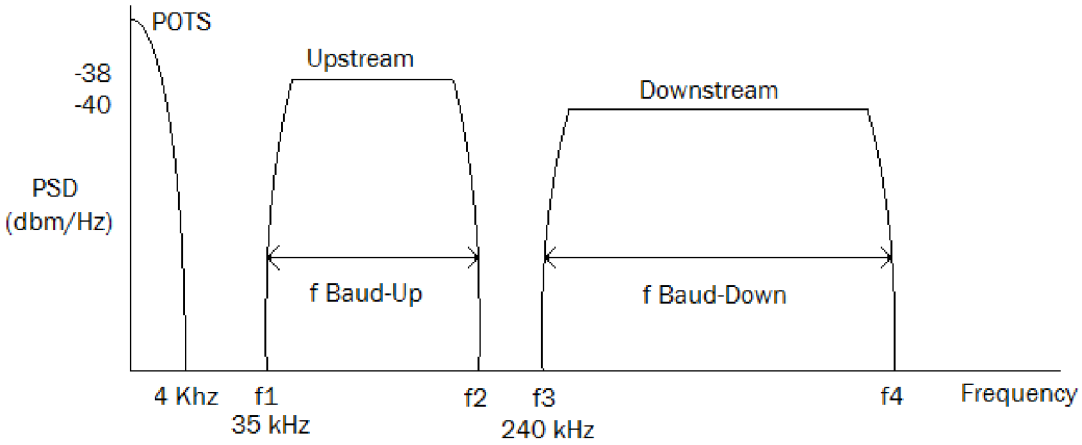


Figure 1.1: CAP Spectrum

CAP modems are able to accept ATM packets, and/or bit synchronous traffic, but as with most ADSL deployments, ATM will predominate across the loop. In addition, an Embedded Operations Channel (EOC) provides for monitoring and troubleshooting of the ADSL modems. Figure 1.2 depicts the data flow for a CAP transmission.

The user data and EOC are injected into the transmission convergence sub-layer, which is responsible for framing and Reed-Solomon encoding and interleaving (Forward Error Correction). The signal then goes through to the physical media dependent element, which performs scrambling, trellis encoding, channel precoding, and the actual CAP transmission.

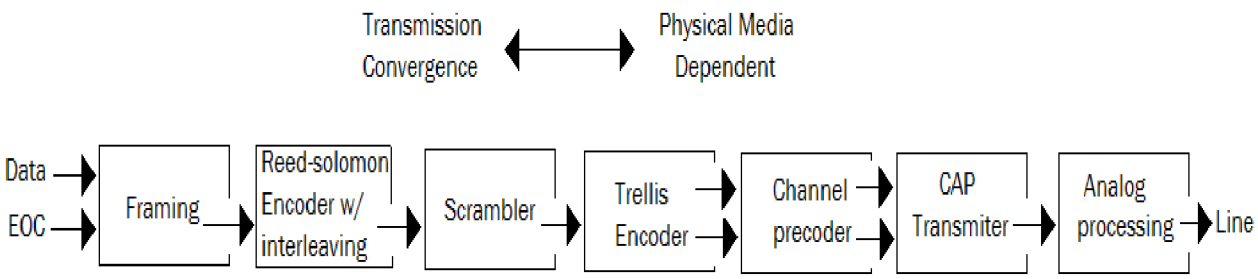


Figure 1.2: the data flow for a CAP transmission

Looking at the data traffic in more detail, CAP in fact defines two types of traffic. The first is Class A, transporting either a packet or cell-based data payload. This channel is not sensitive to latency. A Class B service, however, is designed for latency-sensitive traffic. It may be used to transport a bit synchronous channel like embedded 160 kbps ISDN signal. Class B bypasses FEC which is optional. These two data channels combined with the EOC and are fed into the ADSL modems as depicted in Figure 1.3.

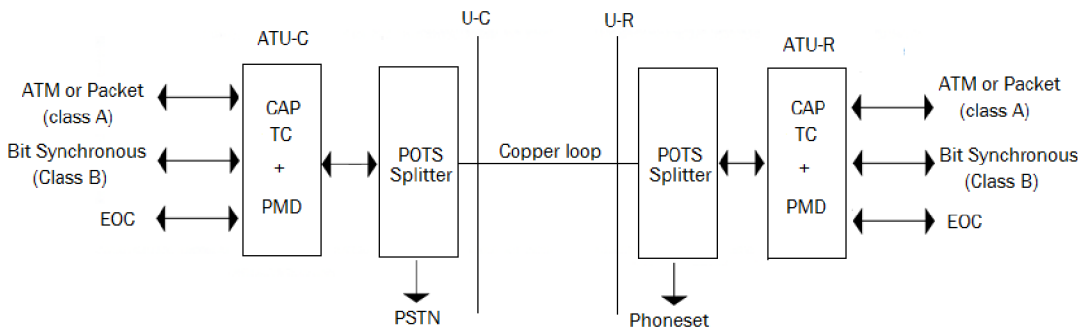


Figure 1.3 CAP Data Types.

Class B, however, is in fact not estimated to be deployed in the major CAP installation, so its fate is preserved. One reason for this is that CAP does not really need a separate path for low-latency traffic since its symbols are transmitted at a high baudrate. This contrasts to the relatively slow transmission of a DMT symbol.

1.5.2 Discrete Multi-tone (DMT)

Although CAP was the encoding of choice for initial ADSL operation, Discrete Multi-tone is now the desired method. DMT encodes the data into a number of narrow subcarriers transmitted at longer time intervals than CAP. As shown in Table **1.4**, DMT consists of 256 subcarriers spaced at 4.3125 kHz.

Subcarrier	Frequency	Meaning
0	0 Hz:	DC - not used for Data
5	25 KHz	Lower limit of upstream data
18	80 KHz	Approx upper limit for 2B'1Q ISDN
28	120 KHz	Approx upper limit for 4B3T ISDN
32	138 KHz	Upper limit of upstream data
64	276 KHz	Pilot - not used for data
256	1 104 KHz	Nyquist - not used for data

Table 1.4: Sub-carrier spacing is 4.3125 kHz - 256 total sub-carriers

The modem may modulate each of the subcarriers at a different bit density which is limited up to a maximum of 15 bits/sec/Hz or 60 kbps/4kHz tone depending upon line noise. For example, at low frequencies where there is less interference, the line may support 10 bits/sec/Hz, while at higher frequencies this may drop to 4 at a corresponding decrease in bandwidth. In extreme cases, subcarriers are shut down due to interference. This use of subcarriers is one of the reasons that DMT is more complicated than CAP in processing requirements, but it has been able to benefit by advances in DSP performance.

Within DMT, 0 kHz (DC) is unused, while the 256th carrier at the Nyquist frequency is not used for data. The lower limit for data traffic in the upstream direction is determined by the POTS/ISDN filters. This also establishes the upstream/downstream frequency split. Finally, a pilot tone modulated to (0,0) is carried within carrier 64 (276kHz). DMT relies on an Inverse Discrete Fourier Transform into each carrier, with the available bandwidth in each function of the number of symbols. This results in a group size of varying complexity up to 256 points.

DMT supports both Synchronous Transfer Mode (STM) and Asynchronous Transfer Mode (ATM) data. In fact, STM was the only mode of operation specified in top priority of the DMT specification. However, since that time, the cell-based ATM option has emerged as the encapsulation of choice, with STM bit synchronous transport the lesser known and unimplemented option.

2 ATM and IP

There is a great debate raging among potential service providers as to whether there should be standard IP–10BT connections or ATM connections to their customer's PCs. The two are very similar—the difference is in the specifics of the equipment and not in the amount of equipment required.

The concern gets more interesting because both architectures usually interface to an ATM backbone network for high-speed connections over a wide area. Therefore, the real issues are the costs of building the network, the services that are to be carried over it, and the time frame for the implementation. If the need is for data services—Internet connections, work at home, etc., the obvious choice is an IP network. The hardware and software required to implement this network is available and relatively inexpensive. There are various advantages to each method of network access:

IP Advantages

- 10BT Ethernet is basically executed by self training.
- Inexpensive LAN PC cards are already manufactured.
- 10BT is an industry standard.
- LAN networks are authenticated and operated in current times .
- There is much expertise in this technology.
- PC software and OS drivers already interface to IP–based LANs.

ATM Advantages

- Streaming video transport has already been authenticated.
- Mixture of services (e.g., video, telephony, and data) is much easier.
- Traffic speeds conform to standard telephony transport rates (e.g., DS–3, STS 1).
- New PC software and drivers will work with ATM.

ATM would be the solution for multiple mixed QoS service requirements in the near future. It is true that the IP technology is being extended to offer layered QoS with RSVP, and IP telephony is being refined to function more effectively. The paradox, however, is that these standards are obsolete today. ATM standards are quite complete. However, not all may be easily implementable. In spite of this, there are many ATM networks in existence or currently under construction.

This leaves the issue of costs. The true costs of creating and operating a large-scale data-access network are unknown. There are portions that are comprehended but many others are only projected. This initiates a great controversy over which

technology is actually less costly. The only way for the costs to be really known is to build reasonably large networks and compare costs. If one technology is a clear winner—a somewhat doubtful hypothesis—then use that technology. If there is no clear cost advantage, then build the network with the service set that matches the service needs of the potential customers. The issue is to start the implementation phase where the real answers will be determined and subsequently end the perpetual discussion phase.

2.1 ATM

Above the copper media and the various ADSL encoding techniques belongs ATM, the data, voice, and video encapsulation of choice for the vast majority of ADSL installations. Further on I shall introduce ATM as a technology followed by physical layer technologies supporting ATM, since ADSL over copper loops is only one of a number of possibilities.

Next it looks at the ATM adaptation layer and finally at the various signalling, traffic management, addressing, and dynamic routing techniques. Note that this basic discussion of ATM is not intended to be exhaustive-the reader is referred to the bibliography contained in the appendix for additional background on what is a very wide subject of discussion.

2.1.1 Concepts and Background

Asynchronous Transfer Mode (ATM) is a technology designed to maintain the quality of service (QoS) requirements of multiple traffic types carried over a single link or network. ATM accomplishes voice, video, and data traffic combination for transport, with bandwidth, loss, latency, and jitter requirements preservation by segmenting all traffic types into 53 byte entities known as cells associated with a different QoS.

Traditional voice traffic is very intolerant of delay and jitter across the network. This traffic will therefore be assigned a QoS guaranteeing proper delivery. In contrast, most data traffic is somewhat tolerant to changes in network performance and may therefore be carried with a less severe QoS. When combining different traffic classes across a single link, the class requiring the more demanding QoS will take priority. Consider an ATM CPE mixing voice and data traffic Figure 2.1.The CPE segments both traffic types into ATM cells, but those belonging to the data traffic will be included in a queue if there is voice traffic to be delivered. On the receiving end, the CPE reassembles the original voice or data frames.

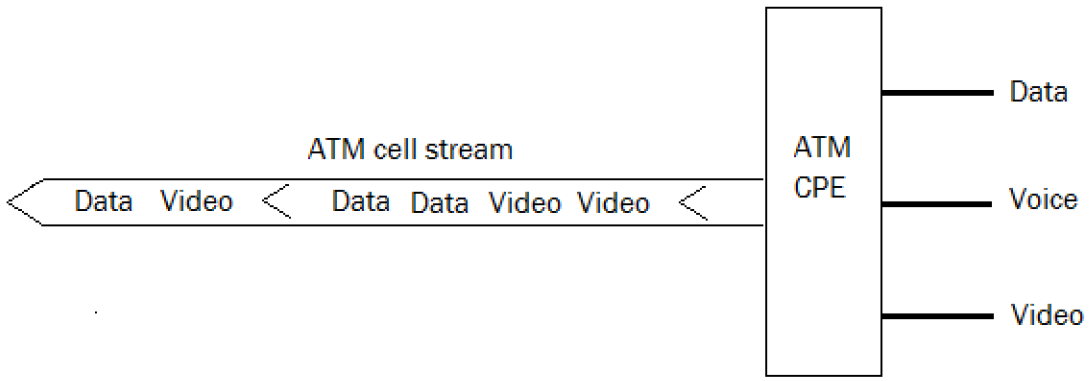


Figure 2.1: ATM CPE mixing voice and data traffic.

ATM is traditionally a connection-oriented technology, establishing a circuit between the source and the destination. The connections might be controlled by the network in the case of Permanent Virtual Circuits (PVCs) or initiated by the subscriber. These Switched Virtual Circuits (SVCs) rely on a signaling protocol. Comparing this to connectionless IP datagram traffic, where the routing protocol routes the packets on-demand across the network. One of the challenges of the last few years has been to integrate properly the connection-oriented ATM Layer with the connectionless IP Layer.

2.1.1.1 ATM Cell Basic Format

ATM transfers information in fixed-size units called cells. Each cell consists of 53 octets, or bytes. The first 5 bytes contain cell-header information, and the remaining 48 contain the payload (user information). Small, fixed-length cells are well suited to transferring voice and video traffic because such traffic is intolerant of delays that result from having to wait for a large data packet to download, among other things. Figure 2.2 illustrates the basic format of an ATM cell.

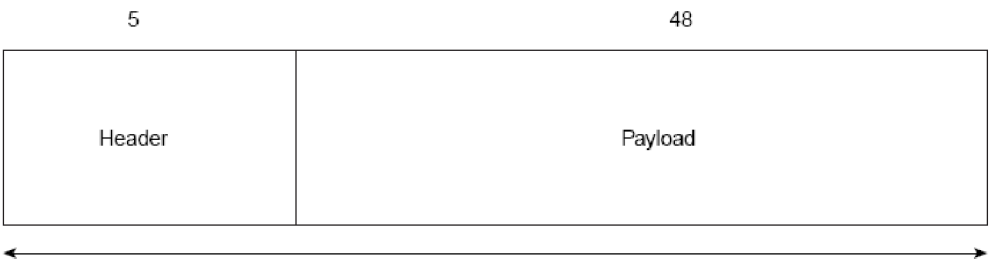


Figure 2.2: an ATM Cell Consists of a Header and Payload Data

2.1.1.2 ATM Cell Header Format

An ATM cell header can be one of two formats: UNI or NNI. The UNI header is used for communication between ATM endpoints and ATM switches in private ATM networks. The NNI header is used for communication between ATM switches. Figure 2.3 depicts the basic ATM cell format, the ATM UNI cell header format, and the ATM NNI cell header format.

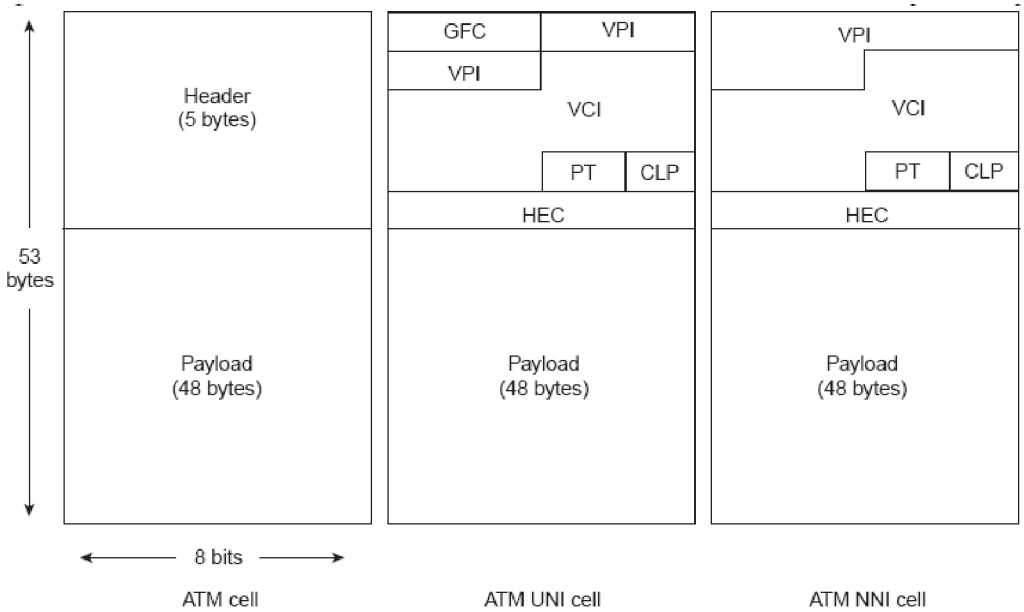


Figure 2.3: An ATM Cell, ATM UNI Cell, and ATM NNI Cell Header Each Contain 48 Bytes of Payload

Unlike the UNI, the NNI header does not include the Generic Flow Control (GFC) field. Additionally, the NNI header has a Virtual Path Identifier (VPI) field that occupies the first 12 bits, allowing for larger trunks between public ATM switches.

2.1.1.3ATM Cell Header Fields

In addition to GFC and VPI header fields, several others are used in ATM cell header fields. The following descriptions summarize the ATM cell header fields illustrated in Figure 2.3:

• **Generic Flow Control (GFC)**—Provides local functions, such as identifying multiple stations that share a single ATM interface. This field is typically not used and is set to its default value of 0 (binary 0000).

• **Virtual Path Identifier (VPI)**—In conjunction with the VCI, identifies the next destination of a cell as it passes through a series of ATM switches on the way to its destination.

• **Virtual Channel Identifier (VCI)**—In conjunction with the VPI, identifies the next destination of a cell as it passes through a series of ATM switches on the way to its destination.

• **Payload Type (PT)**—Indicates in the first bit whether the cell contains user data or control data. If the cell contains user data, the bit is set to 0. If it contains control data, it is set to 1. The second bit indicates congestion (0 = no congestion, 1 = congestion), and the third bit indicates whether the cell is the last in a series of cells that represent a single AAL5 frame (1 = last cell for the frame).

• **Cell Loss Priority (CLP)**—Indicates whether the cell should be discarded if it encounters extreme congestion as it moves through the network. If the CLP bit equals 1, the cell should be discarded in preference to cells with the CLP bit equal to 0.

• **Header Error Control (HEC)**—Calculates checksum only on the first 4 bytes of the header. HEC can correct a single bit error in these bytes, thereby preserving the cell rather than discarding it.

2.1.2.1 ATM Network Interfaces

An ATM network consists of a set of ATM switches interconnected by point-to point ATM links or interfaces. ATM switches support two primary types of interfaces: UNI and NNI. The UNI connects ATM end systems (such as hosts and routers) to an ATM switch. The NNI connects two ATM switches. Depending on whether the switch is owned and located at the customer's premises or is publicly owned and operated by the telephone company, UNI and NNI can be further subdivided into public and private UNIs and NNIs.

A private UNI connects an ATM endpoint and a private ATM switch. Its public counterpart connects an ATM endpoint or private switch to a public switch. A private NNI connects two ATM switches within the same private organization. A public one connects two ATM switches within the same public organization.

An additional specification, the broadband inter-carrier interface (B-ICI), connects two public switches from different service providers. Figure 2.4 illustrates the ATM interface specifications for private and public networks.

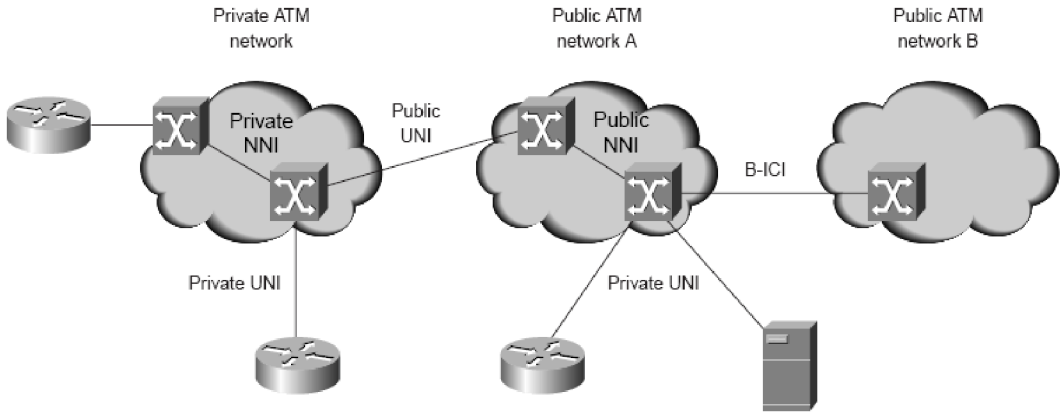


Figure 2.4: ATM Interface Specifications Differ for Private and Public Networks

2.1.2.2 ATM Devices

An ATM network is made up of an ATM switch and ATM endpoints. An ATM switch is responsible for cell transit through an ATM network. The job of an ATM switch is well defined: It accepts the incoming cell from an ATM endpoint or another ATM switch. It then reads and updates the cell header information and quickly switches the cell to an output interface toward its destination. An ATM endpoint (or end system) contains an ATM network interface adapter. Examples of ATM endpoints are workstations, routers, digital service units (DSUs), LAN switches, and video coder-decoders (CODECs). Figure 2.5 illustrates an ATM network made up of ATM switches and ATM endpoints.

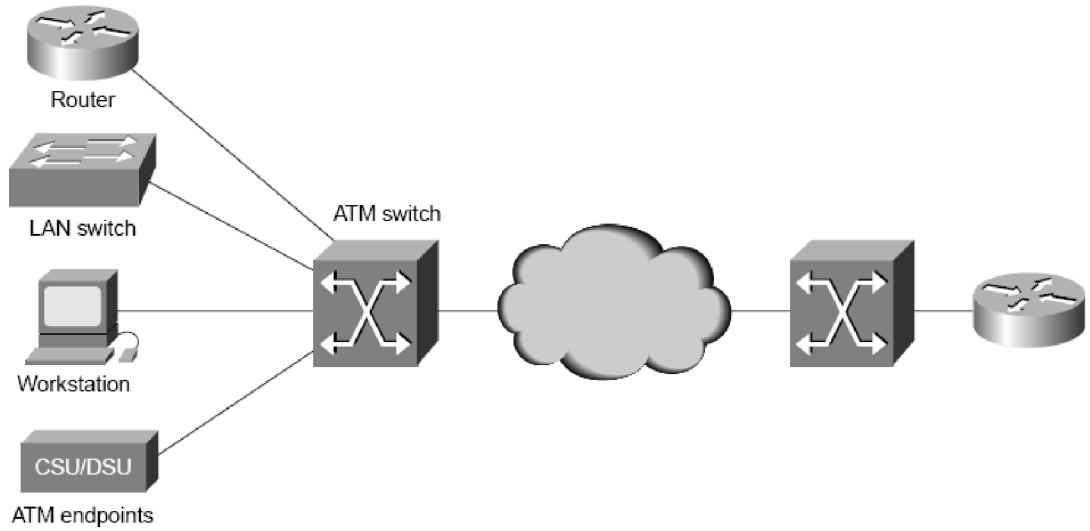


Figure 2.5: An ATM Network Comprises ATM Switches and Endpoints

2.1.2.3 ATM Services

Three types of ATM services exist: permanent virtual circuits (PVC), switched virtual circuits (SVC), and connectionless service (which is similar to SMDS).

PVC allows direct connectivity between sites. In this way, a PVC is similar to a leased line. Among its advantages, PVC guarantees availability of a connection and does not require call setup procedures between switches. Disadvantages of PVCs include static connectivity and manual setup. Each piece of equipment between the source and the destination must be manually provisioned for the PVC. Furthermore, no network resiliency is available with PVC.

An SVC is created and released dynamically and remains in use only as long as data is being transferred. In this sense, it is similar to a telephone call. Dynamic call control requires a signaling protocol between the ATM endpoint and the ATM switch. The advantages of SVCs include connection flexibility and call setup that can be handled automatically by a networking device. Disadvantages include the extra time and overhead required to set up the connection.

2.1.2.3.1 ATM Virtual Connections

ATM networks are fundamentally connection-oriented, which means that a virtual channel (VC) must be set up across the ATM network prior to any data transfer. (A virtual channel is roughly equivalent to a virtual circuit.)

Two types of ATM connections exist: virtual paths, which are identified by virtual path identifiers, and virtual channels, which are identified by the combination of a VPI and a virtual channel identifier (VCI).

A virtual path is a bundle of virtual channels, all of which are switched transparently across the ATM network based on the common VPI. All VPIs and VCIs, however, have only local significance across a particular link and are remapped, as appropriate, at each switch.

A transmission path is the physical media that transports virtual channels and virtual paths. Figure 2.6 illustrates how VCs concatenate to create VPs, which, in turn, traverse the media or transmission path.

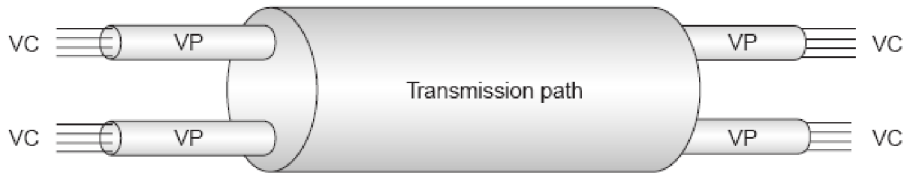


Figure 2.6: VCs Concatenate to Create VPs

2.1.2.3.2 ATM Switching Operations

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

VCIs and VPIs have only local significance across a particular link, these values are remapped, as necessary, at each switch.

2.1.3.1 ATM Reference Model

The ATM architecture uses a logical model to describe the functionality that it supports. ATM functionality corresponds to the physical layer and part of the data link layer of the OSI reference model. The ATM reference model is composed of the following planes, which span all layers:

- **Control**—This plane is responsible for generating and managing signaling requests.
- **User**—This plane is responsible for managing the transfer of data.
- **Management**—This plane contains two components:

  - Layer management manages layer-specific functions, such as the detection of failures and protocol problems.
  - Plane management manages and coordinates functions related to the complete system.

The ATM reference model is composed of the following ATM layers:

• **Physical layer**—Analogous to the physical layer of the OSI reference model, the ATM physical layer manages the medium-dependent transmission.

• **ATM layer**—Combined with the ATM adaptation layer, the ATM layer is roughly analogous to the data link layer of the OSI reference model. The ATM layer is responsible for the simultaneous sharing of virtual circuits over a physical link (cell multiplexing) and passing cells through the ATM network (cell relay). To do this, it uses the VPI and VCI information in the header of each ATM cell.

• **ATM adaptation layer (AAL)**—Combined with the ATM layer, the AAL is roughly analogous to the data link layer of the OSI model. The AAL is responsible for isolating higher-layer protocols from the details of the ATM processes. The adaptation layer prepares user data for conversion into cells and segments the data into 48-byte cell payloads. Finally, the higher layers residing above the AAL accept user data, arrange it into packets, and hand it to the AAL. Figure 2.7 illustrates the ATM reference model.

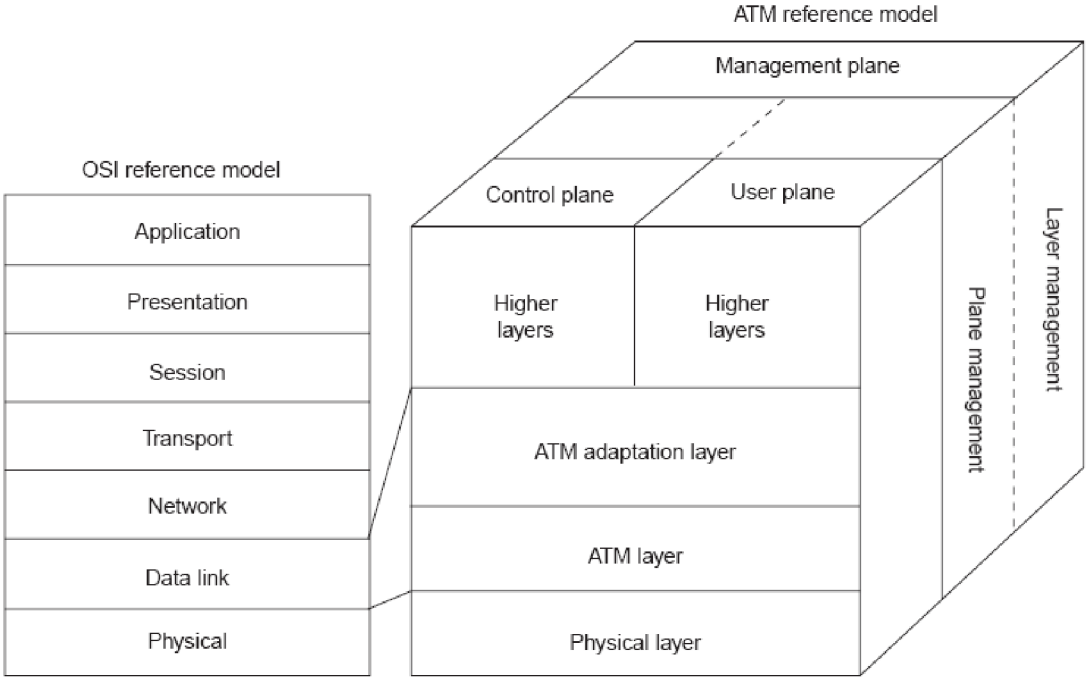


Figure 2.7: The ATM Reference Model Relates to the Lowest Two Layers of the OSI Reference Model.

2.1.3.1.1The ATM Physical Layer

The ATM physical layer has four functions: Cells are converted into a bitstream, the transmission and receipt of bits on the physical medium are controlled, ATM cell boundaries are tracked, and cells are packaged into the appropriate types of frames for the physical medium. For example, cells are packaged differently for SONET than for DS-3/E-3 media types. The ATM physical layer is divided into two parts: the physical medium-dependent (PMD) sub-layer and the transmission convergence (TC) sub-layer.

The PMD sub-layer provides two key functions. First, it synchronizes transmission and reception by sending and receiving a continuous flow of bits with associated timing information. Second, it specifies the physical media for the physical medium used, including connector types and cable. Examples of physical medium standards for ATM include Synchronous Digital Hierarchy/Synchronous Optical Network (SDH/SONET), DS-3/E3, 155 Mbps over multimode fiber (MMF) using the 8B/10B encoding scheme, and 155 Mbps 8B/10B over shielded twisted-pair (STP) cabling.

The TC sub-layer has four functions: cell delineation, header error control (HEC) sequence generation and verification, cell-rate decoupling, and transmission frame adaptation. The cell delineation function maintains ATM cell boundaries, allowing devices to locate cells within a stream of bits. HEC sequence generation and verification generates and checks

The header error control code to ensure valid data. Cell-rate decoupling maintains synchronization and inserts or suppresses idle (unassigned) ATM cells to adapt the rate of valid ATM cells to the payload capacity of the transmission system. Transmission frame adaptation packages ATM cells into frames acceptable to the particular physical layer implementation.

2.2.4.1.2 ATM Adaptation

The following purpose within ATM is data adjustment, out of all layers it is the most sub-layer which attracts attention since the higher layer data types are adapted for transport within the ATM cells and where the actual Segmentation and Reassembly (SAR) is implemented. There are five ATM adaptations defined. The ATM Adaptation Layers (AAL) 1-5 and AAL 2 are usually considered the adaptations for constant bit rate (CBR) and voice traffic (though voice in fact is far from CBR). AAL5 is the data adaptation, while AAL 3 and 4, designed for SMDS/CBDS and for network service providers.

- **ATM Adaptation Layers: AAL1**

AAL1, a connection-oriented service, is suitable for handling constant bit rate sources (CBR), such as voice and videoconferencing. ATM transports CBR traffic using circuit-emulation services. Circuit-emulation service also accommodates the attachment of equipment currently using leased lines to an ATM backbone network. AAL1 requires timing synchronization between the source and the destination. For this reason, AAL1 depends on a medium, such as SONET, that supports clocking.

The AAL1 process prepares a cell for transmission in three steps. First, synchronous samples (for example, 1 byte of data at a sampling rate of 125 microseconds) are inserted into the Payload field. Second, Sequence Number (SN) and Sequence Number Protection (SNP) fields are added to provide information that the receiving AAL1 uses to verify that it has received cells in the correct order. Third, the remainder of the Payload field is filled with enough single bytes to equal 48 bytes. Figure 2.8 illustrates how AAL1 prepares a cell for transmission.

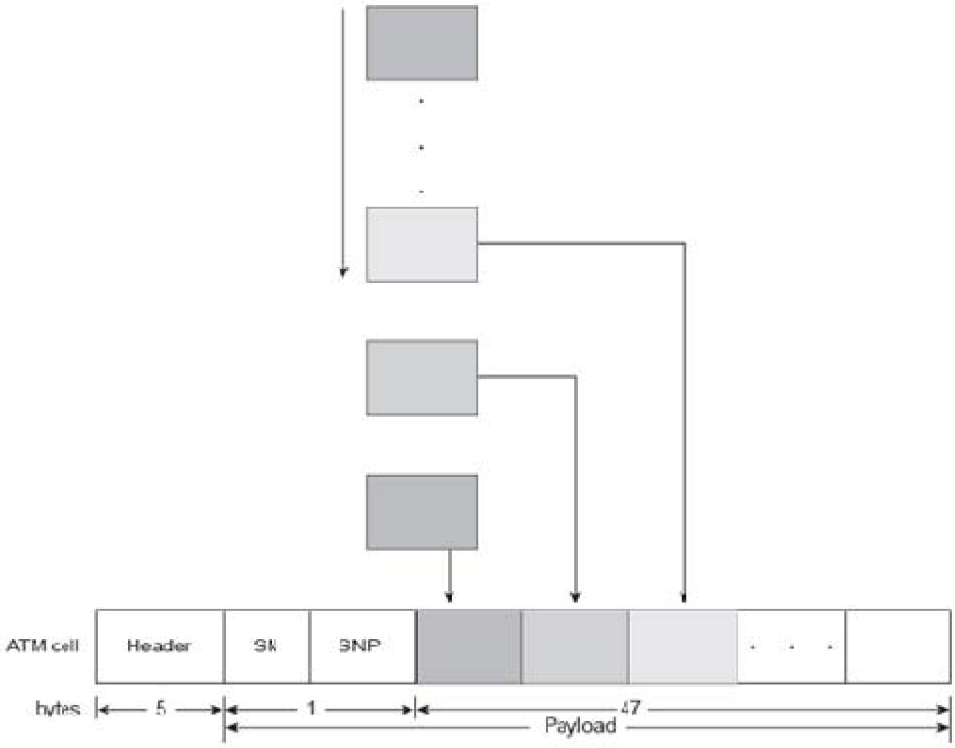


Figure 2.8: AAL1 Prepares a call for transmission.

- **ATM Adaptation Layers: AAL2**

Another traffic type has timing requirements like CBR but tends to be bursty in nature. This is called variable bit rate (VBR) traffic. This typically includes services characterized as packetized voice or video that do not have a constant data transmission speed but that do have requirements similar to constant bit rate services. AAL2 is suitable for VBR traffic. The AAL2 process uses 44 bytes of the cell payload for user data and reserves 4 bytes of the payload to support the AAL2 processes.
VBR traffic is characterized as either real-time (VBR-RT) or as non-real-time (VBR-NRT). AAL2 supports both types of VBR traffic.

- **ATM Adaptation Layers: AAL3/4**

AAL3/4 supports both connection-oriented and connectionless data. It was designed for network service providers and is closely aligned with Switched Multimegabit Data Service (SMDS). AAL3/4 is used to transmit SMDS packets over an ATM network.

AAL3/4 prepares a cell for transmission in four steps. First, the convergence sub-layer (CS) creates a protocol data unit (PDU) by pre-pending a beginning/end tag header to the frame and appending a length field as a trailer. Second, the

segmentation and reassembly (SAR) sub-layer fragments the PDU and pre-pends a header to it. Then the SAR sub-layer appends a CRC-10 trailer to each PDU fragment for error control. Finally, the completed SAR PDU becomes the Payload field of an ATM cell to which the ATM layer pre-pends the standard ATM header.

An AAL 3/4 SAR PDU header consists of Type, Sequence Number, and Multiplexing Identifier fields. Type fields identify whether a cell is the beginning, continuation, or end of a message. Sequence number fields identify the order in which cells should be reassembled. The Multiplexing Identifier field determines which cells from different traffic sources are interleaved on the same virtual circuit connection (VCC) so that the correct cells are reassembled at the destination.

- **ATM Adaptation Layers: AAL5**

AAL5 is the primary AAL for data and supports both connection-oriented and connectionless data. It is used to transfer most non-SMDS data, such as classical IP over ATM and LAN Emulation (LANE). AAL5 also is known as the simple and efficient adaptation layer (SEAL) because the SAR sub-layer simply accepts the CS-PDU and segments it into 48-octet SAR-PDUs without reserving any bytes in each cell.

AAL5 prepares a cell for transmission in three steps. First, the CS sub-layer appends a variable-length pad and an 8-byte trailer to a frame. The pad ensures that the resulting PDU falls on the 48-byte boundary of an ATM cell. The trailer includes the length of the frame and a 32-bit cyclic redundancy check (CRC) computed across the entire PDU. This allows the AAL5 receiving process to detect bit errors, lost cells, or cells that are out of sequence. Second, the SAR sub-layer segments the CS-PDU into 48-byte blocks. A header and trailer are not added (as is in AAL3/4), so messages cannot be interleaved. Finally, the ATM layer places each block into the Payload field of an ATM cell. For all cells except the last, a bit in the Payload Type (PT) field is set to 0 to indicate that the cell is not the last cell in a series that represents a single frame. For the last cell, the bit in the PT field is set to 1.

2.3.1 ATM Connections

ATM supports two types of connections: point-to-point and point-to-multipoint. Point-to-point connects two ATM end systems and can be unidirectional (one way communication) or bidirectional (two-way communication). Point-to-multipoint connects a single-source end system (known as the root node) to multiple destination end systems (known as leaves). Such connections are unidirectional only. Root nodes can transmit to leaves, but leaves cannot transmit to the root or to each other on the same connection. Cell replication is done within the ATM network by the ATM switches where the connection splits into two or more branches.

It would be desirable in ATM networks to have bidirectional multipoint-to-multipoint connections. Such connections are analogous to the broadcasting or multicasting capabilities of shared-media LANs, such as Ethernet and Token Ring. A broadcasting capability is easy to implement in shared-media LANs, where all nodes on a single LAN segment must process all packets sent on that segment.

Unfortunately, a multipoint-to-multipoint capability cannot be implemented by using AAL5, which is the most common AAL to transmit data across an ATM network. Unlike AAL3/4, with its Message Identifier (MID) field, AAL5 does not provide a way within its cell format to interleave cells from different AAL5 packets on a single connection. This means that all AAL5 packets sent to a particular destination across a particular connection must be received in sequence; otherwise, the destination reassembly process will be incapable of reconstructing the packets.

This is why AAL5 point-to-multipoint connections can be only unidirectional. If a leaf node were to transmit an AAL5 packet onto the connection, for example, it would be received by both the root node and all other leaf nodes. At these nodes, the packet sent by the leaf could be interleaved with packets sent by the root and possibly other leaf nodes, precluding the reassembly of any of the interleaved packets.

2.3.1.1 ATM and Multicasting

ATM requires some form of multicast capability. AAL5 (which is the most common AAL for data) currently does not support interleaving packets, so it does not support multicasting.

If a leaf node transmitted a packet onto an AAL5 connection, the packet could be intermixed with other packets and be improperly reassembled. Three methods have been proposed for solving this problem:
VP multicasting, multicast server, and overlaid point-to-multipoint connection.

Under the first solution, a multipoint-to-multipoint VP links all nodes in the multicast group, and each node is given a unique VCI value within the VP. Interleaved packets hence can be identified by the unique VCI value of the source. Unfortunately, this mechanism would require a protocol to uniquely allocate VCI values to nodes, and such a protocol mechanism currently does not exist. It is also unclear whether current SAR devices could easily support such a mode of operation.

A multicast server is another potential solution to the problem of multicasting over an ATM network. In this scenario, all nodes wanting to transmit onto a multicast group set up a point-to-point connection with an external device known as a multicast server (perhaps better described as a resequencer or serializer). The multicast server, in turn, is connected to all nodes wanting to receive the multicast packets through a point-to-multipoint connection. The multicast server receives packets across the point-to-point connections and then retransmits them across the point-to-multipoint connection—but only after ensuring that the packets are serialized (that is, one packet is fully transmitted before the next is sent). In this way, cell interleaving is precluded.

An overlaid point-to-multipoint connection is the third potential solution to the problem of multicasting over an ATM network. In this scenario, all nodes in the multicast group establish a point-to-multipoint connection with each other node in the group and, in turn, become leaves in the equivalent connections of all other nodes. Hence, all nodes can both transmit to and receive from all other nodes. This solution

requires each node to maintain a connection for each transmitting member of the group, whereas the multicast-server mechanism requires only two connections. This type of connection also requires a registration process for informing the nodes that join a group of the other nodes in the group so that the new nodes can form the point-to-multipoint connection.

The other nodes must know about the new node so that they can add the new node to their own point-to-multipoint connections. The multicast-server mechanism is more scalable in terms of connection resources but has the problem of requiring a centralized resequencer, which is both a potential bottleneck and a single point of failure.

2.3.1.2 ATM Quality of Service

ATM supports QoS guarantees comprising traffic contract, traffic shaping, and traffic policing.

A traffic contract specifies an envelope that describes the intended data flow. This envelope specifies values for peak bandwidth, average sustained bandwidth, and burst size, among others. When an ATM end system connects to an ATM network, it enters a contract with the network, based on QoS parameters.

Traffic shaping is the use of queues to constrain data bursts, limit peak data rate, and smooth jitters so that traffic will fit within the promised envelope. ATM devices are responsible for adhering to the contract by means of traffic shaping. ATM switches can use traffic policing to enforce the contract. The switch can measure the actual traffic flow and compare it against the agreed-upon traffic envelope. If the switch finds that traffic is outside of the agreed-upon parameters, it can set the cell-loss priority (CLP) bit of the offending cells. Setting the CLP bit makes the cell discard eligible, which means that any switch handling the cell is allowed to drop the cell during periods of congestion.

2.3.1.3 Data Encapsulations

ATM signaling, routing, addressing, and traffic management are only a support framework for higher-layer services. These services may include voice, video, and data transport. In the latter case, a number of methods of transporting data across the ATM network have been defined, some optimized for the campus and others for the WAN. Those optimized for the campus include LANE and MPOA, while PPP over ATM and MPLS are found in the WAN. RFC-1483 bridging and Classical IP based on RFC-1577, see use in both the campus and the WAN. One of the challenges has been to support higher-layer application requirements across the ATM backbone. Only recently has there been any integration between Layer 3 QoS (IP Precedence, RSVP, and the like) and the ATM service categories. Although these capabilities will evolve in the future, different data models allow this interworking to a lesser or greater extent. They should therefore be considered in the

context of how well they support these requirements in the LAN and WAN, especially as IP-based real-time applications become more commonplace.

2.3.1.3.1 LAN Emulation

LAN Emulation (LANE) is a standard defined by the ATM Forum that gives to stations attached via ATM the same capabilities that they normally obtain from legacy LANs, such as Ethernet and Token Ring. As the name suggests, the function of the LANE protocol is to emulate a LAN on top of an ATM network. Specifically, the LANE protocol defines mechanisms for emulating either an IEEE 802.3 Ethernet or an 802.5 Token Ring LAN. The current LANE protocol does not define a separate encapsulation for FDDI. (FDDI packets must be mapped into either Ethernet or Token Ring-emulated LANs [ELANs] by using existing translational bridging techniques.) Fast Ethernet (100BaseT) and IEEE 802.12 (100VG-AnyLAN) both can be mapped unchanged because they use the same packet formats. Figure **2.9** compares a physical LAN and an ELAN.

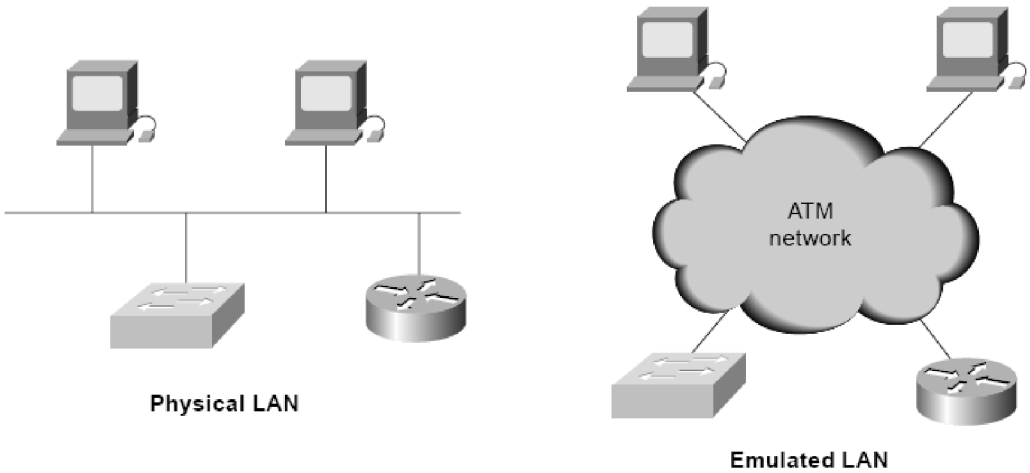


Figure 2.9: ATM Networks Can Emulate a Physical LAN

The LANE protocol defines a service interface for higher-layer (that is, network layer) protocols that is identical to that of existing LANs. Data sent across the ATM network is encapsulated in the appropriate LAN MAC packet format. Simply put, the LANE protocols make an ATM network look and behave like an Ethernet or Token Ring LAN—albeit one operating much faster than an actual Ethernet or Token Ring LAN network.

It is important to note that LANE does not attempt to emulate the actual MAC protocol of the specific LAN concerned (that is, CSMA/CD for Ethernet or token passing for IEEE 802.5). LANE requires no modifications to higher-layer protocols to enable their operation over an ATM network. Because the LANE service presents the same service interface of existing MAC protocols to network layer drivers (such as an NDIS- or ODI-like driver interface), no changes are required in those drivers.

2.3.1.3.2 Multiprotocol over ATM

Multiprotocol over ATM (MPOA) provides a method of transmitting data between ELANs without needing to continuously pass through a router. Normally, data passes through at least one router to get from one ELAN to another. This is normal per-hop routing as experienced in LAN environments. MPOA, however, enables devices in different ELANs to communicate without needing to travel hop by hop.

Figure 2.10 illustrates the process without MPOA in part A and with MPOA in part B. With MPOA enabled devices, only the first few frames between devices pass through routers. This is called the default path. The frames pass from ELAN to ELAN through appropriate routers. After a few frames follow the default path, the MPOA devices discover the NSAP address of the other device and then build a direct connection called the shortcut for the subsequent frames in the flow.

The edge devices that generate the ATM traffic are called multiprotocol clients (MPC) and may be an ATM-attached workstation, or a router. The inter-ELAN routers are called multiprotocol servers (MPS) and assist the MPCs in discovering how to build a shortcut. MPSs are always routers.

This reduces the load on routers because the routers do not need to sustain the continuous flow between devices. Furthermore, MPOA can reduce the number of ATM switches supporting a connection, freeing up virtual circuits and switch resources in the ATM network. Figure 2.10 illustrates the connection before and after the shortcut is established. Note that MPOA does not replace LANE. In fact, MPOA requires LANE version 2.

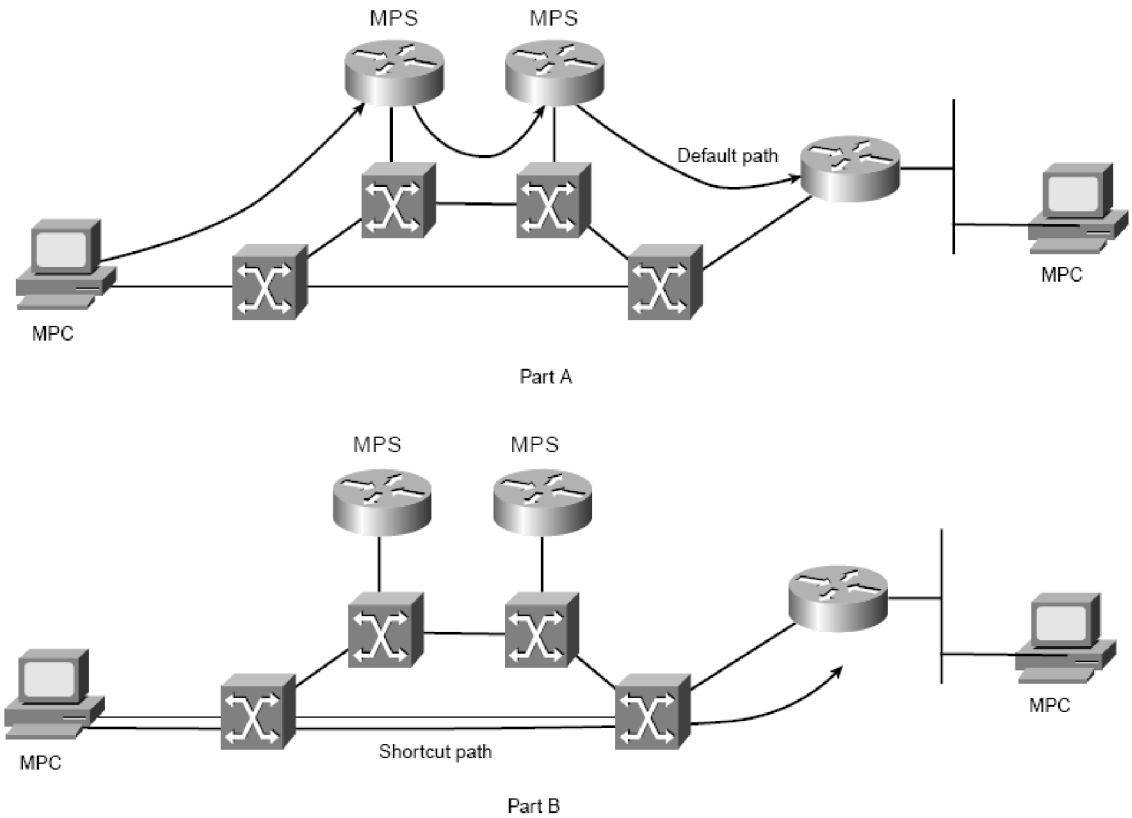


Figure 2.10: Comparison of inter-ELAN Communications Without (Part A) and with (Part B) MPOA.

2.3.1.3.3 RFC-1483

RFC-1483 was defined for the encapsulation routed and/or bridged packets in ATM-AAL-5 cells. The first technique is called LLC/SNAP encapsulation and works with an additional LLC/SNAP header on each packet. This is necessary for the identification of the protocol within the payload filed. The LLC/SNAP header consist of a 3 byte Logical Link Control (LLC), a 3 byte Organisational Unique Identifier (OUI), and a 2byte Protocol Identifier (PID) field. With the PID field every protocol can be distinguished from others. The second technique described in RFC-1483 is called VC multiplexing and differs from LLC/SNAP solution in that the VC is terminated directly at a layer-3 endpoint. This means, the VC-multiplexed connection will carry one protocol only. In a multiprotocol environment, this scheme would use additional VCs. but for the use of IPoATM, the LLC/SNAP technique is the default method, because the UNI signaling required to initiate a LLC/SNAP encapsulated Switched Virtual Connection (SVC). This is defined in RFC-1755. The important advantage its that multiple protocols can share a VC thus limiting the number of VCs required in an IP and multi-protocol environment. On the other hand, it uses an additional 8 byte per AAL frame.

RFC-1483 permanent Virtual Channels (PVCs) between two routers is an effective technique for ATM, having some of the advantages of higher bandwidth and

supporting IP as well as other protocols. This is the reason, why RFC-1483 is also occurred for LANE and MPOA. (3.4)

Under RFC bridging definition, an Ethernet-connected PC encapsulates the IP (or non-IP) traffic in an Ethernet frame and forwards it to the ADSL modem, which encapsulates the Ethernet frame within RFC-1483. The data then passes through the SAR before being transmitted across the ADSL loop. It passes transparently through the DSLAM, finally arriving at a router or LAN switch (Figure 2.11).

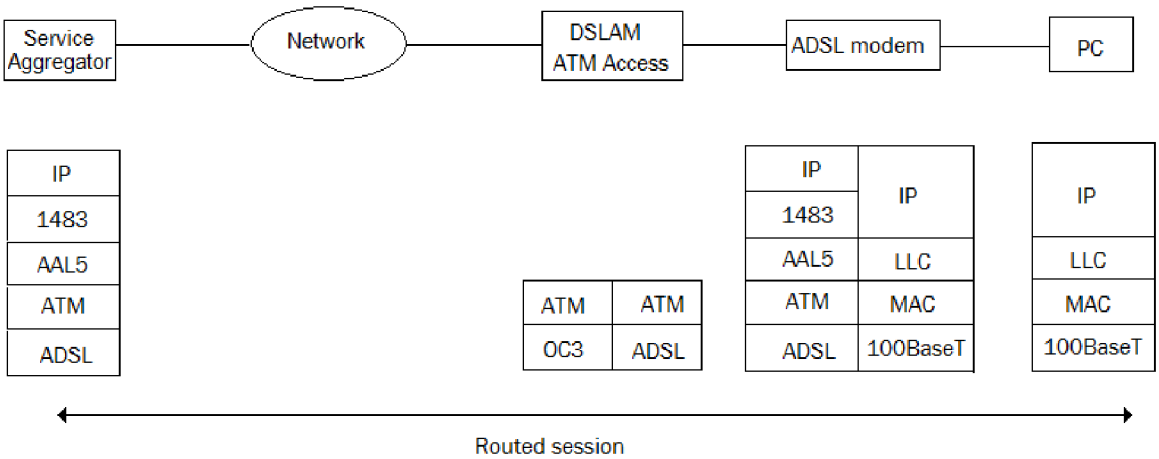


Figure 2.11: RFC-1483 Bridging Protocol Architecture.

If the data arrives at a router, it passes into Layer 3 via an IRB-like function, or alternatively, the router swaps the RFC- 1483 encapsulation for another bridged encapsulation (such as for Frame Relay IAW RFC-2427) before forwarding it on to its ultimate destination. This architecture is preferred by some users who wish to emulate a LAN segment across ADSL. If a LAN switch is deployed instead of a router, an end-to-end bridged encapsulation is all that is available. Note that some degree of security is still possible under this architecture if the router or LAN switch is capable of handling VLANs to segment users or groups of users.

3. Triple play

Triple-play services (voice, video, and data) are emerging as the key driver for telecom investment, with several factors driving this convergence. The Internet is maturing, not only for the transport of web information, but for e-commerce as well. Increasingly sophisticated protocols are emerging to manage and secure the network for business and consumer alike. With this acceptance comes increased scale, ubiquity, and cost effectiveness, far beyond regional telephone and cable

networks. Additionally, the technology to digitally represent, store, and transport all types of data continues to accelerate. Managing these services on a common Internet protocol (IP) network leads to a dramatic reduction in operational expenses.
In the following sections we intend to describe the internet protocol in a simplified manner and the voice over IP with some important issues to provide a good quality for the voice communication. Unfortunately, we won't test the video quality of the IPTV, but there is a big similarity with the VoIP except that the IPTV requires a higher bit-rate.

3.1. INTERNET PROTOCOL (IP)

The Internet Protocol provides the basic unit of data transfer, provides addressing, routing and fragmentation. It is located at the network layer and sends/receives blocks of data called datagrams received from upper layer software. IP feeds these datagrams to its attached data link layer which sends/receives these datagrams as a series of packets altogether. A datagram is analogous to a first-class letter sent in the POST. Over all it will reach its destination but there is no formal authentication that the letter was received like there would be with either a registered or certified mail. IP utilizes a "top effort " or " connectionless" delivery service between source and destination addresses. It is connectionless because there was no formal session initiated between the source and destination before the data was sent. Packets can be lost as they traverse the network or networks; thereby, corrupting datagrams. It is not the responsibility of IP to guarantee the delivery of messages so that IP is frequently termed as an unreliable delivery service. That may be a somewhat harsh of a criticism of IP, but it is the responsibility of the transport layer and not the network layer to guarantee end-to-end message delivery. IP is simply responsible for the addressing and routing of datagrams.

3.1.1 ROUTERS AND HOSTS

Unlike repeaters that functions at the physical layer and bridges the operation of the data link layer. Routers operate at the network layer. A router is used to interconnect two networks together to form an Internet connection. An Internet is a general term used to indicate a group of networks. It is not to be confused with the Internet which is the public network that requires strict addressing standards in order for different systems to communicate. With a control network is kept it completely private and not connect it to the Internet or the corporate Internet or so called Intranet, but if we do we will need a router. IP is a routable protocol and routers are used to implement the protocol. The end-to-end devices on the Internet are called hosts. If two hosts are on the same local network, then messages are routed directly involving no routers. If the two hosts are on different networks, a router must pass the message. This is called indirect routing.

3.1.2 IP ADDRESSING

The IP is responsible for source and destination addresses and its structure is defined in RFC 761. IPv4 is the most frequent version used for addressing and it uses 32-bit addressing. The newer IPv6 calls for 128-bit addressing and was developed because the extended growth of the Internet will soon deplete the inventory of possible 32-bit addresses.

An IP address must not only state a particular host but a particular network as well. The IP address shouldn't be mistaken with the Ethernet II address which is a 48-bit address sometimes called the MAC address. The MAC address is used to facilitate communication only at the data link layer. The IP address facilitates communication over networks and must be universally recognized even if the host is an Ethernet II node attached to a local area network or a serial port attached to a modem. The format of the address is <netid, hostid> but is shown as one 32-bit address split up as four bytes. However, each byte is displayed as a decimal number from 0 to 255. Therefore, an IP address is usually represented as XXX.XXX.XXX.XXX. This address can be shown as a binary or hexadecimal number also, but the decimal-dot-decimal notation is the most popular form. Therefore, the range of addresses is from 0.0.0.0. to 255.255.255.255. An example of an address would be 128.8.120.5, but looking at the address is hard to tell which is the network address and which is the host address. There are five classes of IP addresses: A, B, C, D, E. Class D is for multicasting, a message from one host to many hosts, and class E is reserved for experiments. That leaves classes A, B and C which are the most important. These three classes break up the 32-bit address field into defined address ranges for the netid and hostid. We need to test the very first byte of the IP address to determine the class. If the first bit of this byte is a '0' then this is a class A address. In a class A address the first byte identifies the network and the remaining three bytes identifies the host. That means we can have 16, 277, 214 hosts for every network.

3.1.3 IP HEADER

IP transmits and receives datagrams. Included in the datagram is a header and the data portion of the datagram. The minimum size of the IP header is 20 bytes containing five 32-bit words. The first three words provide control information while the rest of the two words provide address information. An optional field can be subsequent to the address information. The information in the header is shown in figure 3.1 which illustrates The IP datagram consisting of a header and data is inserted into the Ethernet data field.

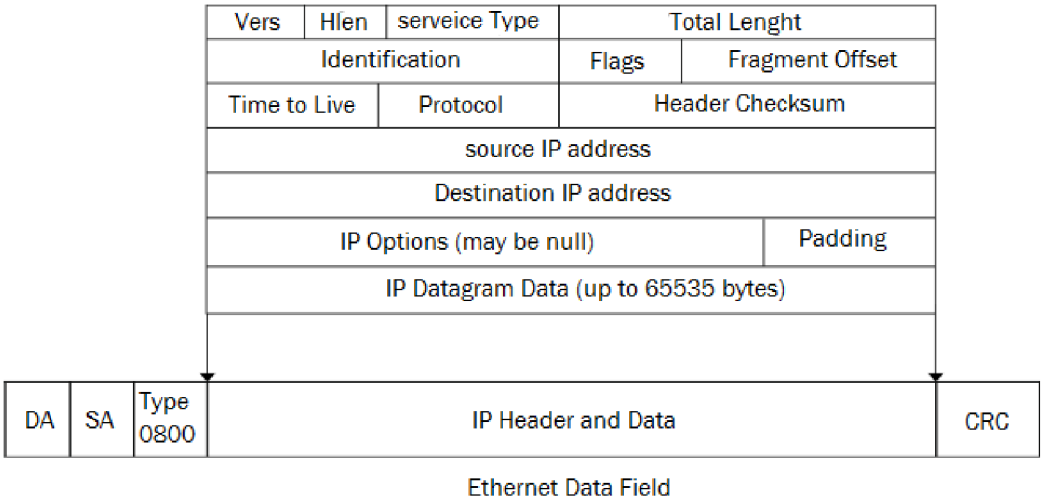


Figure 3.1: The IP datagram consisting of a header and data is inserted into the Ethernet data field.

Version: A four-bit field specifies the IP version. A4 identifies IPv4 while a 6 identifies IPv6.

Header Length: A four-bit field indicates how many four-byte words are in the header. The header length cannot exceed the limit 60 bytes; thereby, allowing 40 bytes for options.

Type of Service: Of the eight-bit field only six bits are used. The Delay bit indicates the datagram should be processed with low delay by the router. The Throughput bit requests high throughput while the Reliability bit requests high reliability. There are three other bits to indicate preference. These bits are set at higher layers of the protocol stack and are suggestions given to the router. This seems like a reasonable feature for control networks since control networks require low delay and high reliability. However, it is unclear that routers even look at these bits. It seems to be that this was a feature with great potential but not executed. This is to be opposed in IPv6.

Total Length: The total length of the datagram including the header cannot reaches more than 65,535 bytes. This 16-bit field is for the datagram itself and not the packet length in the data link layer. If this datagram is larger than the maximum packet length that is allowed to be transferred i.e. sent, then the datagram will need to be fragmented i.e. segmented into manageable successive packets. In this case the total length field will represent the length of the fragment sent and not the length of the original datagram.

Datagram Identification: A unique 16-bit identifier appointed by the host will escort the datagram. This is necessary for the host of reception to gather again fragmented datagrams. All fragments will consist of the same datagram identifier.

Flags: Three bits are reserved for flags but only two are handled. The Don't Fragment bit tells the router not to fragment the datagram. If this cannot be done, an error message is given as a feedback therefore more fragments bit is used in the fragmentation process. A1 means that the datagram being sent is actually a fragment of a larger datagram. A0 means that either the datagram is not fragmented or it's the last fragment. Receiving hosts need this information in order to reassemble fragments.

Fragment Offset: Thirteen bits are used to indicate which fragment is being sent. Fragmentation is the process of breaking up large datagrams into manageable and smaller packets. Remarkably it is required to restrict datagram size to packet size in order to avoid fragmentation. With Ethernet II the maximum packet size is 1500 bytes.

This is called its Maximum Transmission Unit and within a private or local network the MTU is known and can be held to. The problem comes into existence between networks. Intermediate networks may have a lesser MTU requiring the router to fragment the original message even though it was originally sent un-fragmented. The router implements the fragmentation on its own and the fragments must be reassembled at the destination host.

Routers do not recombine fragments. The default MTU is 576 bytes and all routers must be able to handle that size transmission. By limiting the datagram to 576 bytes, it will never need to be fragmented. Sensibly that puts an undue limitation on the Ethernet II network since packets can be as long as 1500 bytes. So for local networks set the maximum datagram size to the local network's MTU. If the datagram is to be sent beyond the local network, set the maximum datagram size to 576 bytes.

For control networks, fragmentation may never be a problem since control information packets are usually short, not exceeding 256 or 512 bytes. Fragmentation should be sidestepped since it increases data latency and increases the opportunities of a corrupted datagram since multiple packets must be sent per datagram. If fragments are to be transferred, it is necessary to load in the fragment offset. Notice that with every fragment, the IP header is resent with just a mere modification.

The fragment offset will change on every fragment and possibly along with one flag bit. Fragments must be sent in eight-byte multiples because there are only 13 bits available for identifying fragments and datagrams can be 64 KB in length. For example, if the first fragment is 1024 bytes long, the fragment offset of the next fragment will indicate that the accompanying fragment begins the 1025th byte of the original datagram.

With knowledge of the datagram identifier, fragment offset, the source IP address and the fragments themselves, the complete datagram can be reassembled by the receiving host even if the fragments are received out of order. That is the true strength of the IP. Packets can take different routes to the intended destination and still be reassembled into the original datagram.

Time to Live: This eight-byte field is strictly used by the routers to prohibit a datagram from a faulty transmission sequence to endlessly circulate around an Internet. Originally, the unit of measure was seconds because it was considered that it would take a router one or more seconds to process a datagram from its queue. Once the datagram was processed, the router would decrement this field by the amount of time that occurred. However, in practice, modern routers are much faster then early routers and usually process the datagram within a second, but only decrement the field by one a minimum amount. Therefore, the field has come to be treated as a hop counter. A hop being is an instance of a datagram which is processed by a router. The originating host sets the Time to Live field and each router decreases it by one. If a router decrements the count to zero, it will remove the datagram and inform the originating host that the datagram failed to reach its destination.

Protocol: The eight-bit protocol field informs the upper layer protocol that the received datagram is for its use. Usually, the upper layer protocol is TCP or UDP, but there are other protocols as well that can handle the sending and receiving of data. The protocol field provides this distinction.

Header Checksum: The complete IP header is checked for its integrity with the aid of the 16-bit header checksum. The originating host applies the checksum and all routers check the header for integrity and reproduce a new checksum when the datagram is resent. Anew checksum is required since the Time to Live field would have been changed by the router. Finally, the checksum is again reconfirmed by the receiving host.

Source/Destination Address: The 32-bit source and destination addresses are included in the header. These are the IP and not MAC addresses.

IP Options: There may be no options in which case this field is null or there can be options usually intended for router use only. The option fields must be at least 32-bits in length and must be padded to that amount if shorter.

3.2 VoIP

For transmission a call through the internet the voice should be converted to digital data (digital signal), this data is compressed and sent over the internet, in packets of 1500 bytes. These packets contain information about their source and destination address, and from this information they will be routed to the receiver and the receiver delivered to the digital data and reconstructed to the original form.

Just like any other data that is sent over the Internet, VoIP data also contains a payload and information that determines where and how the payload will be delivered see Figure 3.2. In VoIP this payload is voice data. The packet also contains other information that helps fast delivery. This allows real time conversations over the Internet.

The Internet Protocol Suite consists of two physical layers - the data link layer and the physical layer. In VoIP the Ethernet is used as the data link layer. This enables reliable transmission of data by controlling and synchronizing the flow. The purpose of the physical layer is to behave as a channel through which information is transferred to the data link layer. Twisted pair cables are used as physical layers in VoIP systems. All network cards, routers, modems, Analog Telephone Adaptors and IP phones are linked to each other through these cables.

Sound signals from the user end are converted to voice packets. These packets are generated via sound cards. After the voice has been converted to digital equivalence, the audio stream is compressed by the VoIP software to enable quick and effective transmission. The compressed data packets consist of all the data required to reach the other end. Thus the quantized data intelligently finds its way through our modem and a maze of twisted cables to the other end of the communication channel.

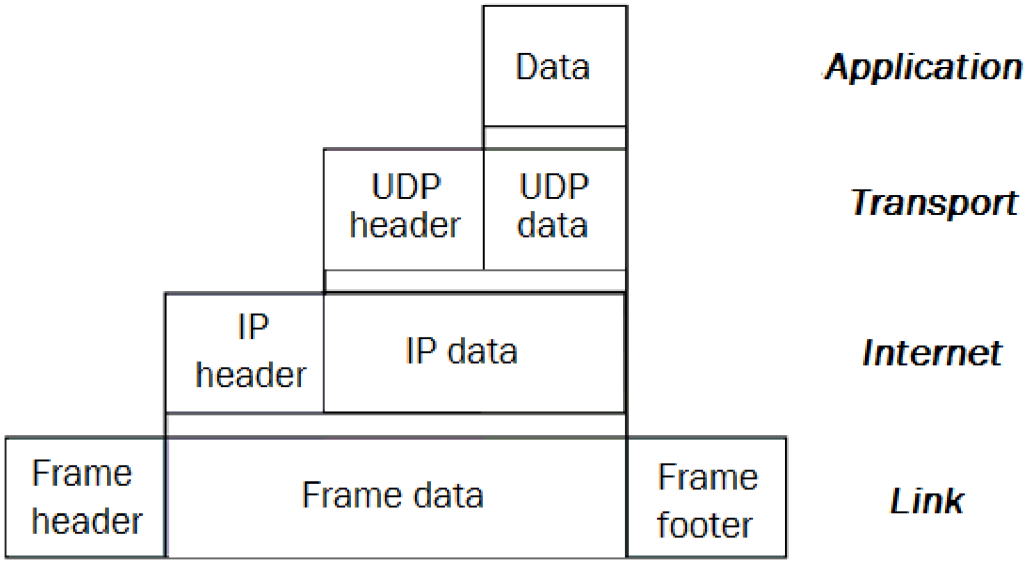


Figure 3.2: data packets covered by information to reach the other end.

The data packets may have to navigate through various paths to reach the intended destination (unlike the ATM Mechanism there should be a virtual circuit establishment). This is because of the transient nature of web traffic. Once the data packets reach the listening end they are arranged in the order in which they were sent and the process of demodulation commences. Here they are converted to some analog equivalent which can be recognized by the listener. The delay or ping time between data transmission and reception must be less than half a second regardless of their locations.

Broadband connection is necessary to communicate over VoIP without realizable delays. This does not introduce any difficulty because increasing number of web surfers are homing in on hi speed connections these days.

3.2.1 Issues of a VoIP Network

There are several issues that need to be addressed in order to provide a toll-quality, PSTN equivalent end-to-end VoIP network. These include:

- Service set to be offered, and the types of end user terminal supported.
- Choice of signaling protocol(s).
- Security.
- Quality of Service (QoS).
- Reliability / availability.
- Regulatory Issues
- Lawful Interception

- Emergency and Operator Services
- Call routing and Number Plans.
- DTMF and Other Tones and Telephony Events
- Firewall and NAT traversal.
- Billing and Reconciliation.
- Network Interconnection.
- Migration Path.
- OSS support.
- Bandwidth Utilization.
- Fax, Modem, and TTY support.
- Auto-configuration.

3.2.1 Service set

A crucial decision facing an operator looking to deploy a VoIP network is the service set that needs to be supported. This could range from a minimal set of services for a “cheap teen line” offering a range of possible broadband data services alongside, through to full PSTN equivalence and advanced services for carriers wishing to replace their current infrastructure with a new converged network for all users.

Another crucial part of the service design is the selection of the end user the terminals that are to be supported by the service offering, possible choices include:

- POTS “black phones”
- IP phones.
- PBXs and key systems
- PC soft-clients (including web-based applications)

IP Phones and PBX systems are located at customer premises and provide voice services. They interact with the Call Agent/SIP Server using a signaling protocol such as SIP, H.323 or a device control protocol such as H.248 (Megaco) or MGCP.

3.2.2 Choice of Signaling Protocols

Numerous different signaling protocols have been developed that are applicable to a VoIP solution. They include:

- Device control protocols such as H.248 (Megaco), MGCP, NCS, etc
- Access services signaling protocols such as SIP, H.323, etc
- Network service signaling protocols such as SIP, SIP-T, BICC, CMSS, etc

The choice of which protocol to use in a service provider network is dependent upon both the service set being offered and the equipment available to provide these services. For example a network must support SIP in order to provide access to SIP phones.

3.2.2.1 The Session Initiation Protocol (SIP)

SIP is a signaling protocol, widely used for setting up and tearing down multimedia communication sessions such as voice and video calls over the Internet. The protocol can be used for creating, modifying and terminating two-party (unicast) or multiparty (multicast) sessions consisting of one or several media streams. The modification can involve changing addresses or ports, inviting more participants, adding or deleting media streams, etc.

The SIP protocol is situated at the session layer in the OSI model, and at the application layer in the TCP/IP model. SIP is designed to be independent of the underlying transport layer; it can run on TCP, UDP, or SCTP. SIP has the following characteristics: Transport-independent, because SIP can be used with UDP, TCP, SCTP, etc. Text-based, allowing for humans to read and analyze SIP messages. Protocol design SIP clients typically use TCP or UDP (typically on port 5060 and/or 5061) to connect to SIP servers and other SIP endpoints. SIP is primarily used in setting up and tearing down voice or video calls. However, it can be used in any application where session initiation is a requirement. These include Event Subscription and Notification, Terminal mobility and so on. There are a large number of SIP-related RFCs that define behavior for such applications. All voice/video communications are done over separate session protocols, typically RTP.

3.2.3 Quality of Service

One of the key requirements for the widespread deployment of VoIP is the capability to offer a toll quality service equivalent to the existing PSTN. Indeed some carriers are even looking for Next-Generation Networks as a mean for delivering much higher voice quality as a service.

Perceived Voice quality is very sensitive to three key performance criteria in a packet network, in particular:

- Delay
- Jitter
- Packet loss

IP, by its nature, gives out the best-effort service and does not provide guarantees about the key criteria. Therefore it is necessary to apply an appropriate QoS solution in the majority of cases where simple over provisioning cannot guarantee success. There are a large number of technologies that can be chosen to offer QoS support such as Diffserv, RSVP,MPLS and even ATM. However the objective of such a solution is always to guarantee prioritization of voice media streams over best-effort data, and to ensure that the voice service is not compromised by unforeseen traffic patterns.Jitter is a specific VoIP Quality of Service issue that may affect the quality of the conversation if it goes out of control.

Unlike network delay, jitter does not occur because of the packet delay, but because of a variation of packet delays. As VoIP endpoints try to compensate for jitter by increasing the size of the packet buffer, jitter causes delays in the conversation. If the variation becomes too high and exceeds 150ms, callers notice the delay and often revert to a walkie-talkie style of conversation.

There are several steps to be taken to reduce jitter both on the network level and in the VoIP endpoints such as VoIP software, IP phones or dedicated VoIP adaptors. By definition, reducing the delays on the network helps keep the buffer under 150ms even if a significant variation is present. While the reduced delay does not necessarily remove the variation, it still effectively reduces the degree to which the effect is pronounced and brings it to the point where it's unnoticeable b the callers.Prioritizing VoIP traffic and implementing bandwidth shaping also helps reduce the variation of packet delay.

At the endpoint, it is essential to optimize jitter buffering. While greater buffers reduce and remove the jitter, anything over 150ms noticeably affects the perceived quality of the conversation. Adaptive algorithms to control buffer size depending on the current network conditions are often quite effective. Fiddling with packet size or using a different codec (e.g. G.711) often helps control jitter.

Packet loss occurs in every kind of network. All network protocols are designed to cope with the loss of packets in one way or another. TCP protocol, for example, guarantees packet delivery by sending re-delivery requests for the lost packets. RTP employed by the VoIP protocol does not provide delivery guarantee, and VoIP must implement the handling of lost packets.

While a data transfer protocol can simply request re-delivery of a lost packet, VoIP has no time to wait for the packet to arrive. In order to maintain call quality, lost packets are substituted with interpolated data.

A technique called Packet Loss Concealment (PLC) is used in VoIP communications to mask the effect of dropped packets. There are several techniques that may be used by different implementations.

The R-factor metrics in VoIP, called R-factors, use a formula to take into account both user perceptions and the cumulative effect of equipment impairments to arrive at a numeric expression of voice quality. In its simplest form and in the range of 0 - 100, the R-factor can be calculated as:

$$\text{R-factor} = R_o - I_s - I_d - I_{e\text{-}eff} + A \qquad (3.1)$$

Where R_o is a basic Signal-to-Noise Ratio. I_s is all impairments that occur more or less simultaneously with the voice signal. I_d is Delay impairment factor. $I_{e\text{-}ef}$ is Effective equipment impairment factor caused by low bit-rate codec and by packet loss on the network path. A is the advantage factor, which is a compensator to off-set the others. For example using a DECT handset or a mobile phone gives user

mobility advantages which off-set the impairment caused by (for example) errors in the radio interface

VoIP calculates two equipment impairment values to report as voice quality metrics: the Network R-factor and the User R-factor. The Network R-factor is generated based on the physical equipment impairments. The User R-factor adds perceptual effects to the equipment impairment, such as recency and delay. The user R-factor attempts to add the "perceived" annoyance that a user may experience during a call based on a perceptual effect called recency.

Recency is an auditory phenomenon where distracting events that have occurred more recently appear to have a greater impact on perceived quality. The User R-factor has been found to match well with users purely subjective ratings of voice quality.

These metrics are calculated by a formula that balances all equipment impairments and perception factors. Each metric is reported as a single number on a per-call basis, typically in the range of 15 to 94. Lower numbers indicate greater equipment impairment or perceived poor voice quality. In VoIP, calls are broken down into a set of ranges for the Network R-factor and User R-factor values calculated for each call. The actual R-factor numbers associated with a single call can be viewed in the Channel Details Table for the call.

In general, the R-factors should map well to a sliding scale of how voice quality is perceived. At the extremes, calls with values greater than 80 will have few quality problems and those with values less than 50 will have significant problems. The Network R-factor can be compared to the User R-factor to help determine which factors predominate in any voice quality degradation -- equipment impairments such as packet loss, or, more subjective factors such as recency and delay. The table 3.3.shows ranges of voice quality for the R-factors.

The R-factor is also converted to a Mean Opinion Score (MOS), which corresponds to purely subjective rating (Figure 3.4) by users of speech quality on a numeric scale of 1 to 5.

Desirability Scale	R-factor	MOS Value
Desirable	94 – 80	4.4 – 4.0
Acceptable	80 – 70	4.0 – 3.6
Reach Connection	70 – 50	3.6 – 2.6
Not Recommended	50 – 0	2.6 – 0

Table 3.3 The ranges of voice quality for the R-factors

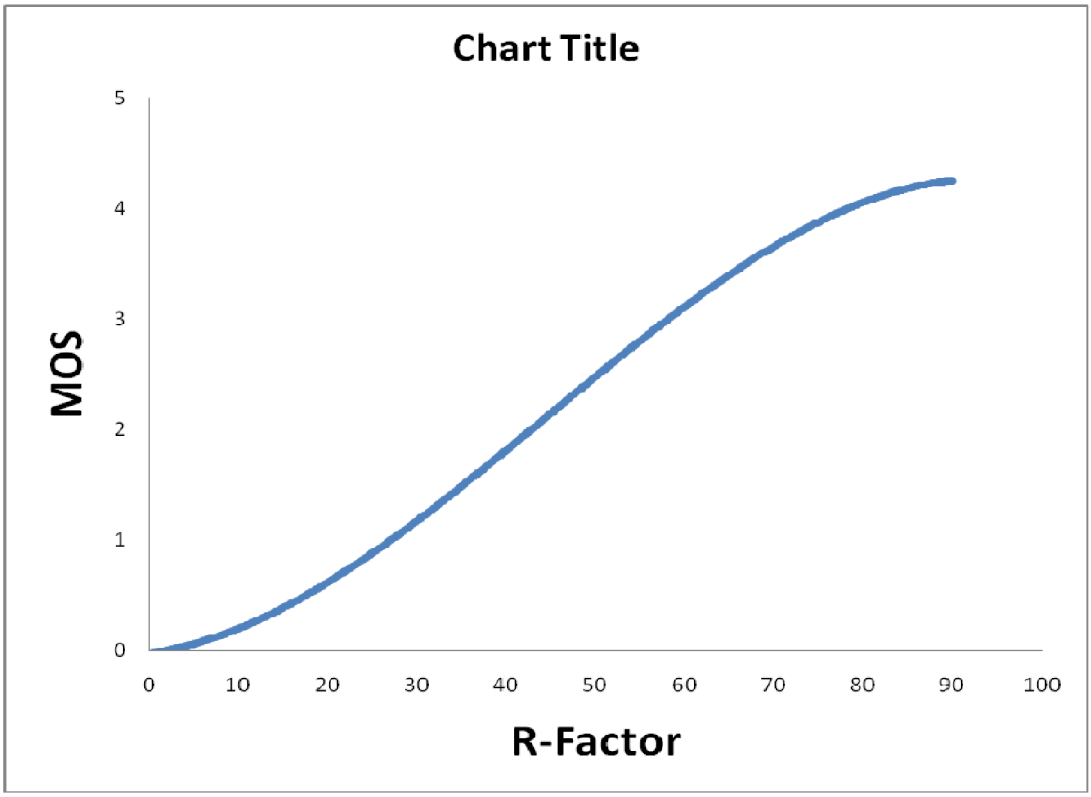


Figure 3.4: the relation between MOS and R-factor.

3.2.4 Reliability / Availability

The PSTN achieves five-nines reliability, equivalent to fewer than five minutes per year downtime, and it handles millions of simultaneous calls. A VoIP network needs to achieve similar levels of reliability and scalability.

This can be achieved in a VoIP network by using redundant and load allocating equipment and networks. The call agent, access gateway, trunk gateway, signaling gateway and media server need to be fault tolerant. The types of functionality often used to achieve fault tolerance include:

- Redundant hardware
- Redundant network connections
- Hot-swap capability
- No single point of failure
- Software and firmware that can be upgraded without loss of service

3.2.5 Call Routing and Number Plans

The PSTN enables the routing of calls between telephones anywhere in the world. This is achieved by having a well-defined number plan both nationally and internationally. Routing tables can be built using this numbering plan to provide end-to-end connectivity.

A Next-Generation VoIP Network must bestow the same capability, which requires the following:

- International and National numbering /addressing plans, for example ENUM implementations
- Interconnection to the PSTN and E.164 numbers
- SIP endpoint addressing schemes
- Allocation of numbers/addresses and number portability issues
- Call routing between numbers/addresses

3.2.6 DTMF and Other Tones and Telephony Events

When using VoIP there is an issue in transporting DTMF and Other Tones and Telephony Events. These can flow transparently using a full rate code such as G.711 but can't be transported using lower-bit codecs such as G.729.

There are several solutions used for transporting these tones and events but the most widespread are:

- The usage of RTP packets as specified by RFC.
- The transportation of the DTMF tones out of band using the signaling protocol, e.g. SIP or H.248.

G.711 is a common open source and royalty free, high bit rate codec. This codec does not require licensing fees and uses very little computational resources while providing the best possible sound quality at the expense of higher than usual network bandwidth. On the other hand, G.723 and G.729 (patent protected in some countries) consume 3 to 4 times less bandwidth than G.711 at the expense of increased CPU and memory load and slightly lower sound quality. There are numers other free and licensed codecs on the market, each offering a different tradeoff between computational requirements, bandwidth, and voice quality.

3.2.7 Firewall and NAT traversal

For equipment which are accommodated in customer premises, such as IP phones and Subscriber Gateways are likely that a firewall will be established at the edge of the customer premises. In addition, Network Address Translation may be used to convert internal IP addresses to external IP addresses.

Therefore it is important that both the RTP media traffic and the signaling flows (SIP, H.248, MGCP) can negotiate both NAT and the firewall. For the firewall to be effective it requires to ensure that only authorized flows enter or leave the networks.

There are working groups within the IETF, including Midcom and NSIS, who are addressing the issue of communications with firewalls and network address translators.

3.3 IPTV

Just as we have seen voice go through digitizing in the 60's and eventual packetized delivery over IP networks, we will soon see the same happening with TV. The technologies developed for VoIP, video conferencing, and IP in general will be applied to an emerging technology that will provide a more function-rich, user-interaction form of TV to the consumer.

The phone companies will most likely be the first to embrace the IPTV technology, since this will help open up a new market for them. The added functionality realized through the capabilities of a packetized, two-way TV system will undoubtedly attract a lot of customers. This will put pressure on the CATV industry to adopt similar functions in their systems, which will most likely lead to IPTV over cable as well. Cable companies have used this packetized IP technology to deliver telephony services over CATV for several years now using VoIP The phone companies are already delivering packet-based IP traffic over their DSL-capable customer connections. This is the basis for providing DSL customers with Internet access. Therefore, the move to IPTV is a natural choice. The primary roadblocks to deployment of IPTV over DSL deal with bandwidth and sufficient quality of service (QoS). The goal is to provide the customer with TV quality equal to that delivered by CATV. That's mean the service provider needs to offer the maximum number of applications possible to their users.

3.3.1 Video Doubles Communications ARPU

Technological advances was the development of DSL which was mentioned previously. It was the development of DSL that allowed a POTS line to let its lower bandwidth to be used for voice, while the higher bandwidth was redistributed for Internet protocol high speed packet data. Initially most home users started subscribing to 1.5 Mbps (megabits per second) DSL service. However, even at the beginning stages of DSL service, knowledgeable communication workers knew it was capable of 6 Mbps delivery over short distances (18kft or less). The initial 6 Mbps at the beginning stages of DSL service was capable of transporting a DVD (Digital Versatile Disk) quality TV signal which requires only 4 Mbps. Additionally, as information, to understand the bandwidth required for some of these other communication technologies, the article Telcos Take on Cable with Video Delivery (2004) describes the bandwidth needed for various applications by stating, "it requires 2 Mbps to transmit a broadcast quality video stream using MPEG-2 compression. A DVD-quality TV signal takes 4 to 5 Mbps; and HDTV (High Definition Television) requires approximately 9 Mbps." Therefore at the outset of DSL it could not transmit a HDTV signal at 9 Mbps, but it was capable of transporting a 4 to 5 Mbps DVD-quality TV signal although other convergent technologies were needed to

support the application of IPTV delivery that were not ready for public distribution to the ordinary household.

3.3.2 Transmission

The subject of convergent technologies takes us to the next pieces of the puzzle required for IPTV delivery. Just as the development of fiber optic rings were already in existence and then more pieces of the puzzle were added with DSL, so has the development of convergent technological innovations of equipment capable of receiving TV and HDTV satellite signals at the service provider's head-end locations.

This equipment at the service provider's head-end location is capable of compressing and then converting the incoming signals into packets to send out over Telecommunication Company's high speed SONET (Synchronous optical network) fiber optic rings. These TV and HDTV signals can now be transported through the fiber optic rings and distributed to central locations throughout the service provider's telecommunications networks to get to the individual consumers local loop.

In addition to the above technologies being developed to receive these TV and HDTV signals at the service provider's head-end locations, there has to be something at the end users locations (the consumer) to be able to receive these packet signals from the DSL line and convert them into usable human information, such as the TV signal. That technology is a set top box that accepts signals such as Internet DSL, IPTV delivering standard TV and HDTV, VOIP, and standard POTS. This brings us to the subject of the next technological innovation. As we have discussed, DSL at its beginning stages could only deliver 6 Mbps. However, to do all the technologies listed above, DSL will certainly have to increase it's bandwidth of 6 Mbps. The new DSL technologies that are capable of delivering much higher bandwidths over a single copper pair are called ADSL2+ and VDSL. Figure 3.4 illustrates the differences that the various DSL technologies provide in bandwidth and physical loop length over a single copper pair (5).

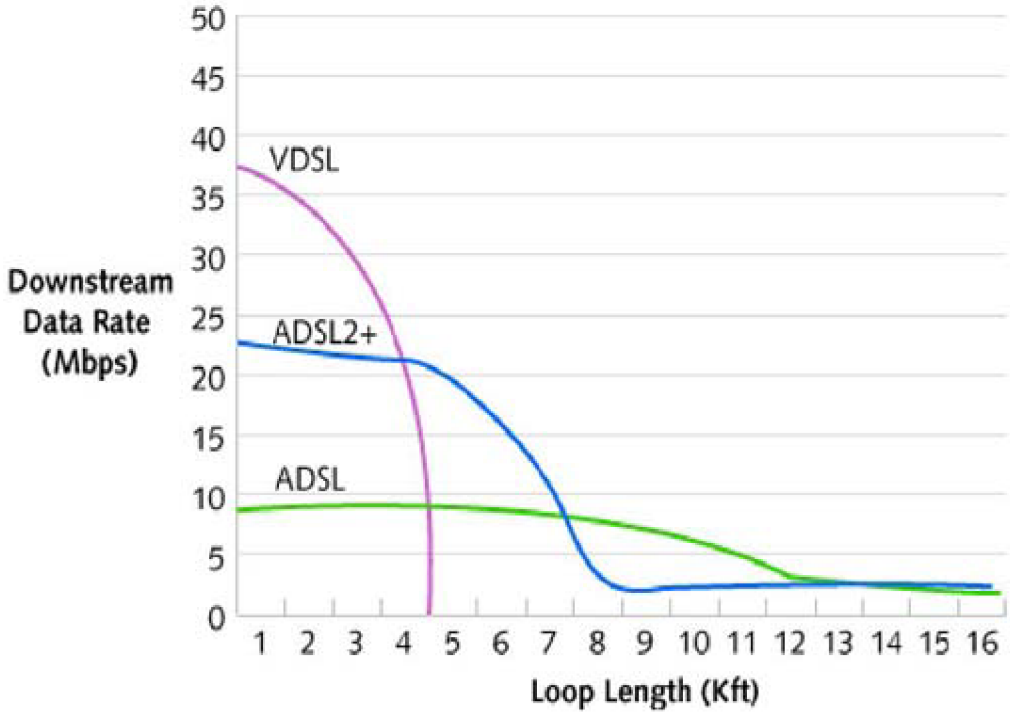


Figure 3.4: the differences between the various DSL technologies provide in bandwidth and physical loop length.

However, it is important to note when looking at the chart above, that various techniques are being tried to improve the bandwidth and the physical loop length capabilities of these various DSL technologies. One of those techniques is bonding copper cable pairs from the Service Provider's Central Offices or the Service Provider's remote SLC terminal cabinets to the consumer. Bonding is the process of terminating 2 copper cable pairs together to act as 1 larger copper cable pair in order for it to carry larger bandwidth capabilities. O'Shea(6) states that bonding allows two or more DSL lines to be aggregated into one virtual line and that when an ADSL2+ line is bonded, it can carry as much as 45 Mbps bandwidth. The article also states that most telcos will be initially focusing on this type of ADSL2+ bonded technology.

3.3.3 Codecs

Assessing the performance of various codecs is often a largely objective operation since different codecs perform better for some content than they do others. Probably the best test for IPTV usage is "DVD Quality" video or "HDTV Quality". Even within these categories, performances can vary significantly depending on the type of material, like action, animated, etc. In addition, determining which compression technique leads to a 'better picture' can be quite objective.

Some of the potential codecs today are the MPEG's (MPEG2 and MPEG4), the 'H-series' (H.264), Sorenson3, RealVideo 9 and 10, and Windows Media Video 9 (and 10 is on the drawing board). Microsoft sees IPTV as a viable strong new technology

and is getting into it in a number of ways. They are currently going into trials with SBC. SBC sells long distance, DSL Internet service, and phone and data products and services, and currently serves more than 52 million access lines nationwide. Basically, what Microsoft is offering is an "IPTV in a box" for companies that aren't in the television business right now.

According to most sources, the Microsoft codecs do not appear to be leading the pack in terms of best compression and quality, although the verdict is still out. Most recent studies have selected the RealVideo and the Sorenson as the superior codecs for video. However, as history has demonstrated in the past, it is not always the best that becomes the de facto standard, but rather the most aggressive. Figure 3.5 shows a performance comparison for 90-minute DVD-quality movie at 700 Kbps(17).

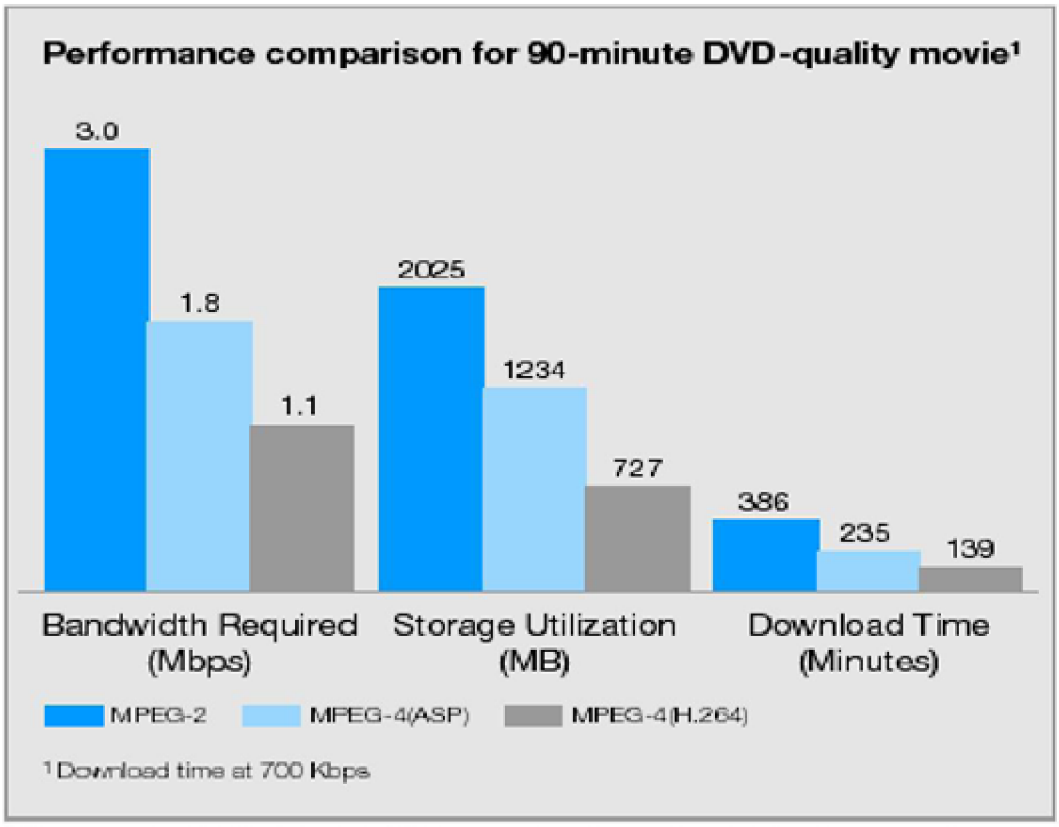


Figure 3.5: Shows a performance comparison for 90-minute DVD-quality movie at 700 Kbps.

Utilizing these new encoding and compression techniques means to the service provider the HDTV delivered signal that required 9 Mbps bandwidth utilizing MPEG-2 technology, will now only take 5.5 Mbps bandwidth utilizing MPEG-4 technology and only 5 Mbps bandwidth with WMV-9 (16). It means the once standard-definition broadcast bandwidth requirement of 4 Mbps using MPEG-2, will now only require 1.5

Mbps with WMV-9(4). This means the service provider utilizing the most advanced encoding and compression technologies can almost deliver two HDTV channels over the same amount of bandwidth that one HDTV channel required utilizing MPEG-2 technology. Therefore, the service providers that are able to utilize these newer compression technologies will be able to offer far more content delivery using less bandwidth, which will strategically enhance their ability to compete.

3.3.4 IPTV Architecture

IPTV delivery is very different from Cable TV delivery. Cable TV architecture is designed to deliver all the broadcast content channels from their head-end locations simultaneously over a single transport feed with a defined bandwidth capability to large serving areas such as an entire city. This signal carrying all the broadcast content channels is then delivered throughout the city or large serving areas with coax or fiber to the individual neighborhoods and then coax drops are brought into each individual consumers home. On cable TV, the consumer is receiving the entire line-up of broadcast content channels all the time which requires huge amounts of bandwidth being delivered all the time into the home.

As more advanced telecommunications services and applications requiring more bandwidth per channel become available to consumers, such as a wider listing of HDTV available channels, the legacy cable TVs' single transport feeds will not have the bandwidth needed to deliver these services. Cable TV service providers are presently faced with either growing their infrastructure to accommodate these greater bandwidth needs to meet consumer demands or converting their existing cable architecture systems to IPTV based architecture systems. Either way, they are strategically facing huge amounts of capital investment in order to stay competitive in the future. Unlike cable TV delivery architecture, IPTV architecture is very different in that it only delivers the single channel that is requested by the consumer's individual TV all the way from the IPTV service provider's head-end equipment. Therefore, with IPTV the infrastructure needed to support huge amounts of bandwidth being delivered all the time is not needed.

The infrastructure for IPTV service providers only needs to support the specific request for channel bandwidth that is requested from the consumer at any given time.

In an IPTV environment, everything is, in a sense, an on-demand stream out to the set-top box. We may have 250 channels available on our IPG (Interactive Program Guide), but they're not being broadcast 24/7. Instead, the set-top box tunes to a channel, and a separate stream is then sent down to that box. So, in a DSL IPTV environment, we can have an unlimited number of channel choices and an unlimited number of VOD titles available. Our limitation is not how fat the pipe is; it's how many servers we (meaning the service provider) have and what our business model is in terms of how broad a choice we want to offer. For the case of the CATV, the limit will

depend on the size of the pipe and on the number of customers that will be demanding service at one time.

For example, CATV might have 100 channels @6M/channel = 600M total BW. At a codec rate of 5M/feed, that means 120 customers can be served at one time on that cable system.

4. Monitoring softwares

4.1 Wireshark

Wireshark is a network packet analyzer. A network packet analyzer will try to capture network packets and tries to display that packet data as detailed as possible. We could think of a network packet analyzer as a measuring device used to examine what's going on inside a network cable, just like a voltmeter is used by an electrician to examine what's going on inside an electric cable .Wireshark is perhaps one of the best open source packet analyzers available today.

Here are some examples people use Wireshark for:

- network administrators use it to **troubleshoot network problems**
- network security engineers use it to **examine security problems**
- developers use it to **debug protocol implementations**
- people use it to **learn network protocol** internals

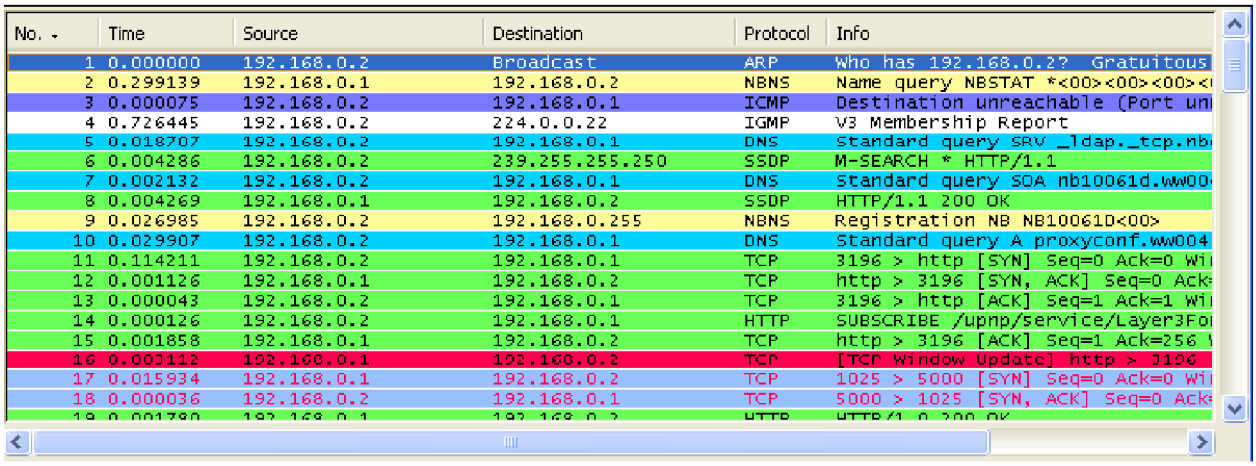
Beside these examples, Wireshark can be helpful in many other situations too.

The following are some of the many features Wireshark provides:

- Available for **UNIX** and **Windows**.
- **Capture** live packet data from a network interface.
- Display packets with **very detailed protocol information**.
- **Open and Save** packet data captured.
- **Import and Export** packet data from and to a lot of other capture programs.
- **Filter packets** on many criteria.
- **Search** for packets on many criteria.
- **Colorize** packet display based on filters.
- Create various **statistics**.
- ... and **a lot more!**

4.1.1 The "Packet List" pane

The packet list pane displays all the packets in the current capture file illustrated in Figure 4.1:



No.	Time	Source	Destination	Protocol	Info
1	0.000000	192.168.0.2	Broadcast	ARP	Who has 192.168.0.2? Gratuitous
2	0.299139	192.168.0.1	192.168.0.2	NBNS	Name query NBSTAT *<00><00><00><(
3	0.000075	192.168.0.2	192.168.0.1	ICMP	Destination unreachable (Port un
4	0.726445	192.168.0.2	224.0.0.22	IGMP	V3 Membership Report
5	0.018707	192.168.0.2	192.168.0.1	DNS	Standard query SRV _ldap._tcp.nb
6	0.004286	192.168.0.2	239.255.255.250	SSDP	M-SEARCH * HTTP/1.1
7	0.002132	192.168.0.2	192.168.0.1	DNS	Standard query SOA nb10061d.ww00
8	0.004269	192.168.0.1	192.168.0.2	SSDP	HTTP/1.1 200 OK
9	0.026985	192.168.0.2	192.168.0.255	NBNS	Registration NB NB10061D<00>
10	0.029907	192.168.0.2	192.168.0.1	DNS	Standard query A proxyconf.ww004
11	0.114211	192.168.0.2	192.168.0.1	TCP	3196 > http [SYN] Seq=0 Ack=0 Wi
12	0.001126	192.168.0.1	192.168.0.2	TCP	http > 3196 [SYN, ACK] Seq=0 Ack
13	0.000043	192.168.0.2	192.168.0.1	TCP	3196 > http [ACK] Seq=1 Ack=1 Wi
14	0.000126	192.168.0.2	192.168.0.1	HTTP	SUBSCRIBE /upnp/service/Layer3Fo
15	0.001858	192.168.0.1	192.168.0.2	TCP	http > 3196 [ACK] Seq=1 Ack=256
16	0.001112	192.168.0.1	192.168.0.2	TCP	[TCP Window Update] http > 3196
17	0.015934	192.168.0.1	192.168.0.2	TCP	1025 > 5000 [SYN] Seq=0 Ack=0 Wi
18	0.000036	192.168.0.2	192.168.0.1	TCP	5000 > 1025 [SYN, ACK] Seq=0 Ack

Figure 4.1: The "Packet List" pane.

Each line in the packet list corresponds to one packet in the capture file. If we select a line in this pane, more details will be displayed in the "Packet Details" and "Packet Bytes" panes.

While dissecting a packet, Wireshark will place information from the protocol dissectors into the columns. As higher level protocols might overwrite information from lower levels, we will typically see the information from the highest possible level only.

For example, let's look at a packet containing TCP inside IP inside an Ethernet packet. The Ethernet dissector will write its data (such as the Ethernet addresses), the IP dissector will overwrite this by its own (such as the IP addresses), the TCP dissector will overwrite the IP information, and so on.

There are a lot of different columns available. Which columns are displayed can be selected by preference settings.

The default columns will show:

- **No.** The number of the packet in the capture file. This number won't change, even if a display filter is used.
- **Time** The timestamp of the packet. The presentation format of this timestamp can be changed.
- **Source** The address where this packet is coming from.
- **Destination** The address where this packet is going to.
- **Protocol** The protocol name in a short (perhaps abbreviated) version.
- **Info** Additional information about the packet content. There is a context menu available by clicking right mouse click.

4.1.2 Capturing Live Network Data

Capturing live network data is one of the major features of Wireshark. The Wireshark capture engine provides the following features:

- Capture from different kinds of network hardware (Ethernet, Token Ring, ATM, ...).
- Stop the capture on different triggers like: amount of captured data, captured time, captured number of packets.
- Simultaneously show decoded packets while Wireshark keeps on capturing.
- Filter packets, reducing the amount of data to be captured.
- Capturing into multiple files while doing a long term capture, and in addition the option to form a ring buffer of these files, keeping only the last x files, useful for a "very long term" capture.

The capture engine still lacks the following features:

- Simultaneous capturing from multiple network interfaces (however, we can start multiple instances of Wireshark and merge capture files later).
- Stop capturing (or doing some other action), depending on the captured data.

4.1.3 The "Capture Interfaces" dialog box

When we select "Interfaces..." from the Capture menu, Wireshark pops up the "Capture Interfaces" dialog box as shown in Figure 4.2**:**

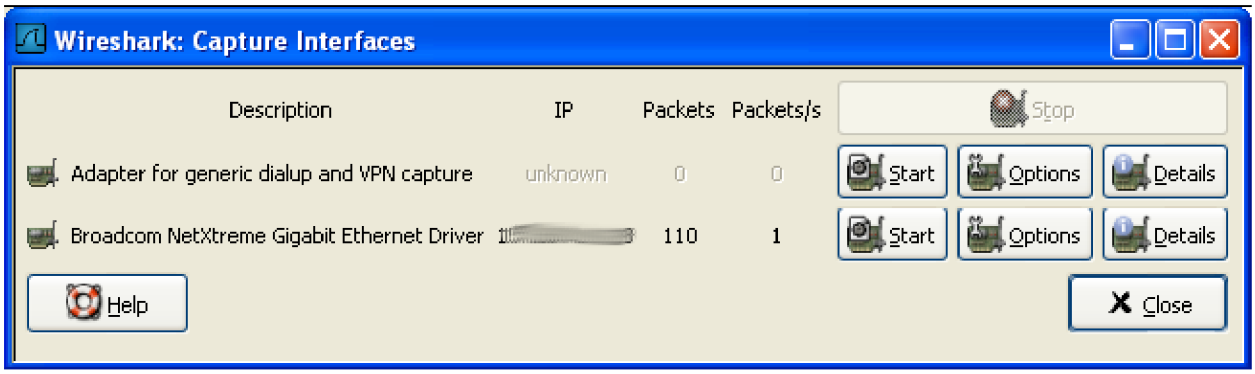


Figure 4.2: The "Capture Interfaces" dialog box

Description. The interface description provided by the operating system.

IP. The first IP address Wireshark could resolve from this interface. If no address could be resolved (e.g. no DHCP server available), "unknown" will be displayed. If more than one IP address could be resolved, only the first is shown (unpredictable which one in that case).

Packets. The number of packets captured from this interface, since this dialog was opened. Will be greyed out, if no packet was captured in the last second.

Packets/s. Number of packets captured in the last second. Will be greyed out, if no packet was captured in the last second.

Stop. Stop a currently running capture.

Start. Start a capture on this interface immediately, using the settings from the last capture.

Options .Open the Capture Options dialog with this interface selected.

Details. Open a dialog with detailed information about the interface.

Help. Show this help page.

Close. Close this dialog box.

4.1.2.1 While a Capture is running.

While a capture is running, the following dialog box is shown below in Figure 4.3:

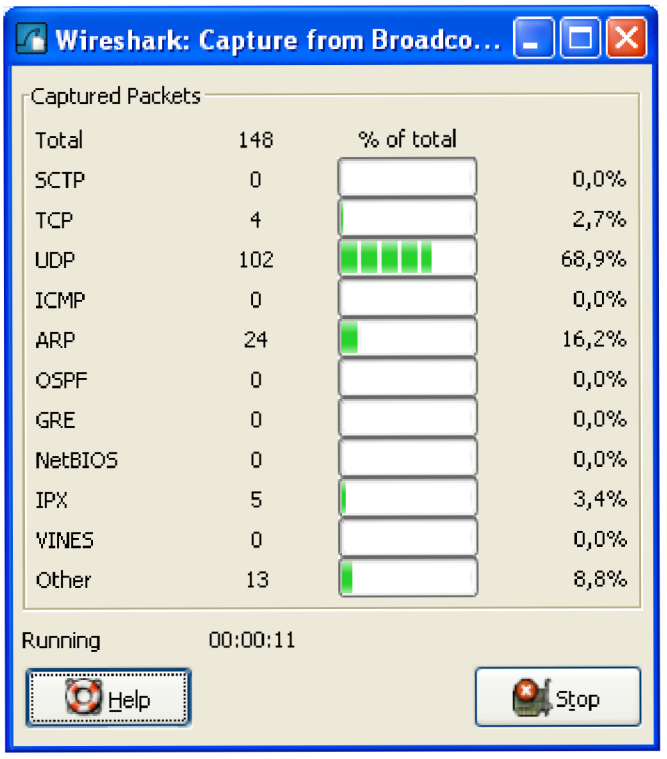


Figure 4.3: While a capture is running

This dialog box will inform us about the number of captured packets and the time since the capture was started. The selection of which protocols are counted cannot be changed

4.1.2.2 Finding packets

We can easily find packets once we have captured some packets or have read in a previously saved capture file. Simply we select the **Find Packet...** menu item from the **Edit** menu. Wireshark will pop up the dialog box shown in Figure 4.4.

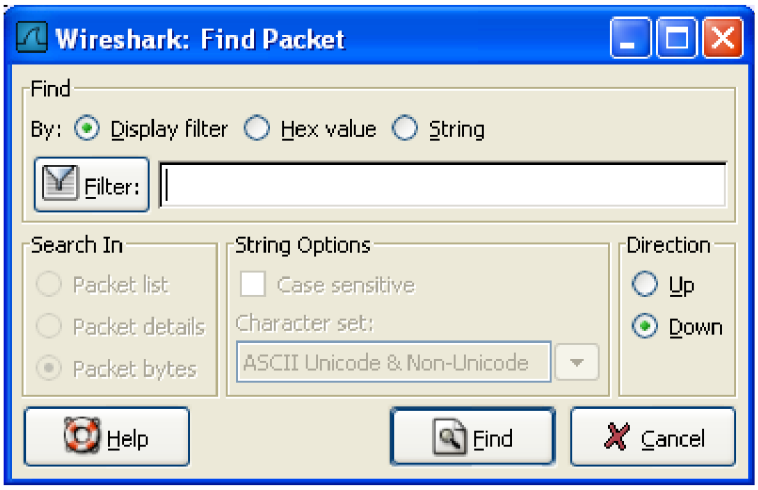


Figure 4.4: The "Find Packet" dialog box

We might first select the kind of thing to search for:

- **Display filter** Simply enter a display filter string into the **Filter:** field, select a direction, and click on OK.

For example, to find the three way handshake for a connection from host 192.168.0.1, use the following filter string: ip.src==192.168.0.1 and tcp.flags.syn==1 For more details on display filters.

- **Hex Value** Search for a specific byte sequence in the packet data. For example, use "00:00" to find the next packet including two null bytes in the packet data.

- **String** Find a string in the packet data, with various options. The value to be found will be syntax checked while we type it in. If the syntax check of our value succeeds, the background of the entry field will turn green, if it fails, it will turn red.

We can choose the search direction:

- **Up** Search upwards in the packet list (decreasing packet numbers).
- **Down** Search downwards in the packet list (increasing packet numbers).

4.1.2.3 Filtering packets while viewing

Wireshark has two filtering languages: One used when capturing packets, and one used when displaying packets. We will explore the second type of filtering: Display

filters. Display filters allow us to concentrate on the packets we are interested in while hiding the currently uninteresting ones. They allow us to select packets by:

- Protocol
- The presence of a field
- A comparison between fields
- and many more!

To select packets based on protocol type, we simply type the protocol in which we are interested in the "**Filter**" field in the filter toolbar of the Wireshark window and press enter to initiate the filter. Figure 4.5 "Filtering on the TCP protocol" shows an example of what happens when we type **tcp** in the filter field.

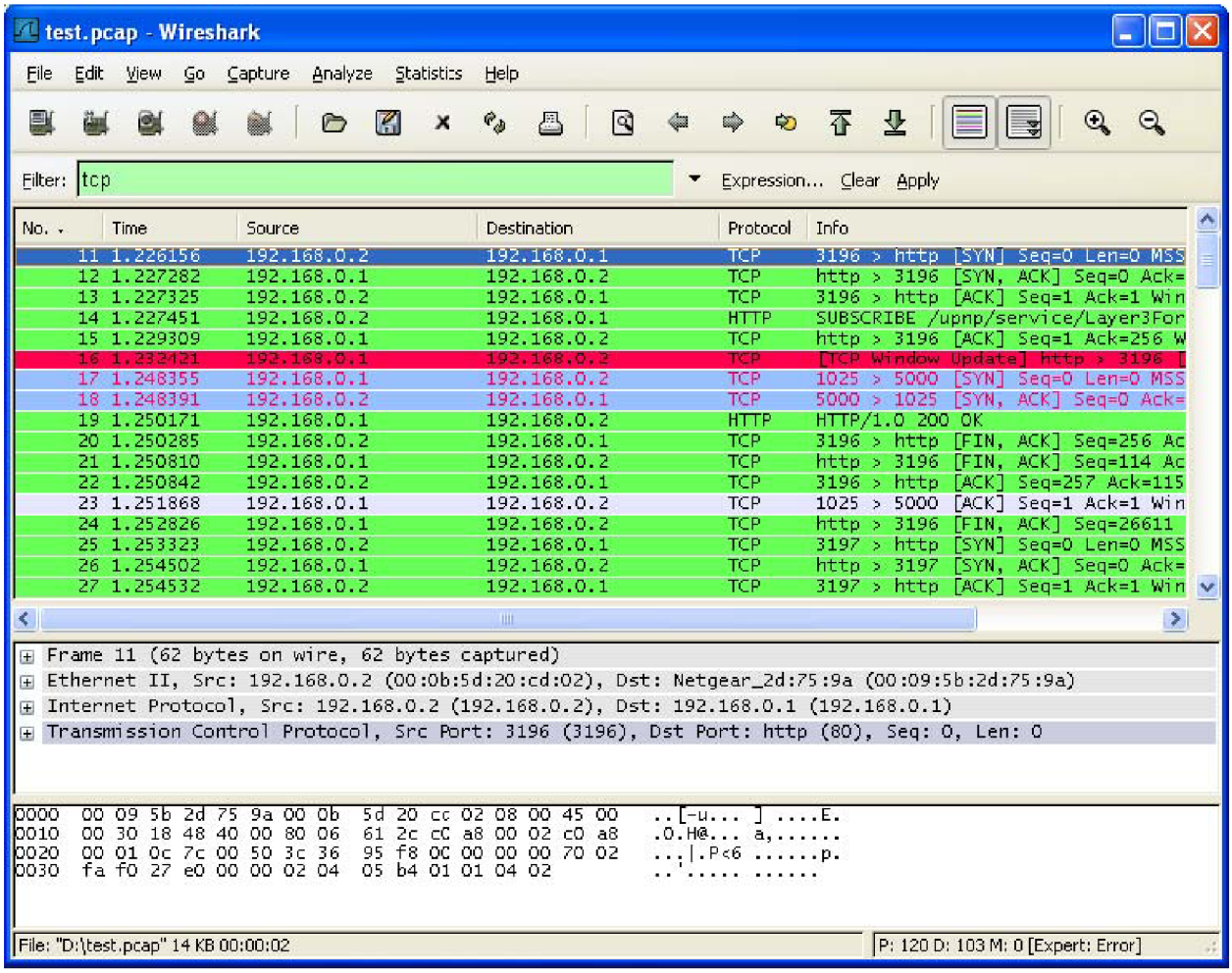


Figure 4.5: **Filtering on the TCP protocol**

The packets of the TCP protocol are displayed now (e.g. packets 1-10 are hidden). The packet numbering will remain as before, so the first packet shown is now packet number 11. We can filter on any protocol that Wireshark understands. We can also filter on any field that a dissector adds to the tree view, but only if the dissector has added an abbreviation for the field. A list of such fields is available in Wireshark in the **Add Expression...** dialog box.

For example, to narrow the packet list pane down to only those packets to or from the IP address 192.168.0.1, use **ip.addr==192.168.0.1**.

4.2 OmniPeek

OmniPeek voice and video analysis derives its call quality metrics from industry-standard Telchemy technology. Voice over IP and Video over IP refer to protocol suites used to set up and maintain two way voice or video communications over the Internet. Voice and video protocol suites include those relating to SIP, SCCP, RTSP, etc. The unit of communication is the call and an individual call may be carried in multiple channels, some dedicated to signaling and others to carrying the encoded voice data. The encoded data is referred to as media, and a call containing such data has media channels. Media channels contain RTP (Real-time Transport Protocol) or RTCP (RTP Control Protocol) data. The conversion of voice data into digital form and back again is accomplished using a particular codec (coder/decoder), specified in the RTP header.

The Voice and Video views in capture windows provide simultaneous analysis of voice and video traffic with subjective and objective quality metrics. The Calls view displays one row for each call in a capture and the Media view displays one row for each RTP media flow in a call. The Voice and Video Visual Expert displays signal bounce diagrams of the signaling and RTP/RTCP packets of an entire call in a single window.

4.2.1Voice and Video view window

The Voice and Video views have two data areas which are illustrated in Figure 4.6. The upper pane contains voice and video data arranged by call or by the media streams within a call.

The lower pane contains three tabs which represent additional information for a row or rows selected in the upper pane, allowing us to view call details, a summary count of the Voice and Video expert events found in the capture or a capture log of the individual VoIP expert events. The parts of the Voice and Video view window are identified below.

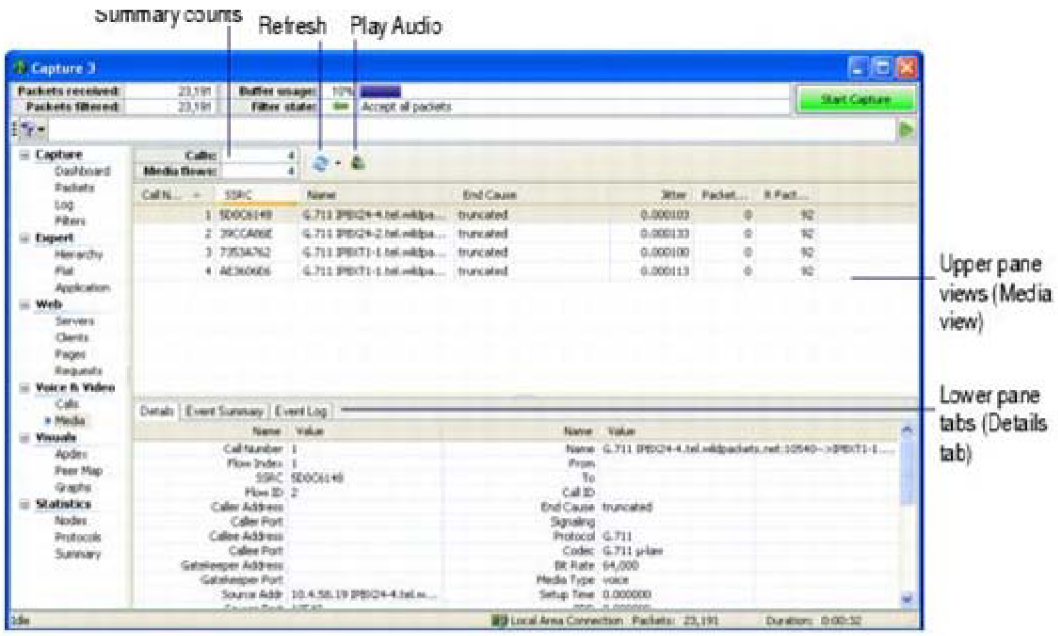


Figure 4.6: Voice and Video view window.

Summary counts: This area displays the total calls and media flows in this capture.

Refresh: We can immediately update the currently displayed Voice & Video view with the latest information. We can also choose a refresh interval from the drop-down list. **Play Audio**: This button leads us to play the audio from a call or media flow that has a playback-supported codec. The button is only available when a selected call or media flow has a playback-supported codec.

Upper pane Voice and Video views: This area displays voice or video data arranged by calls or media.

Lower pane Voice and Video tabs: This area displays additional information corresponding to a selected row of data in the upper pane.

4.2.1.1 Voice and Video upper pane views

The upper pane contains captured voice data arranged in two formats: by individual call or by the individual media streams within a call.

4.2.1.1.1 Calls view

The Calls view in Figure 4.7 displays one row for each call. Each call is displayed in the order in which it was captured, with call number, call name, and end cause information.

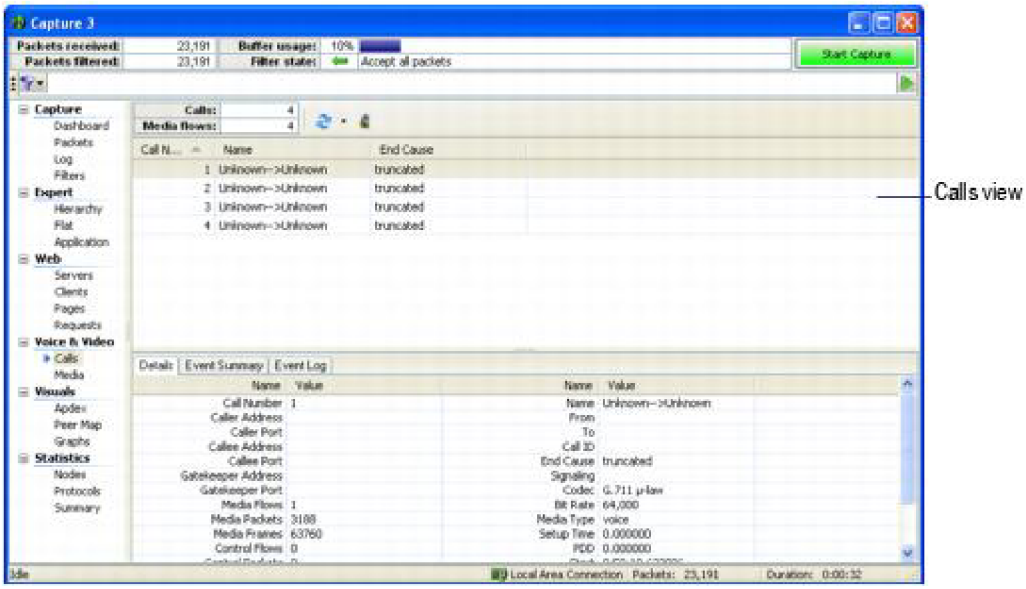


Figure 4.7: the Calls view.

4.2.1.1.2 Media view

The Media view in Figure 4.8 displays one row for each RTP media flow per call. A voice call will usually have two media flows, one for each direction. Video calls will usually have four media flows: two voice and two video.

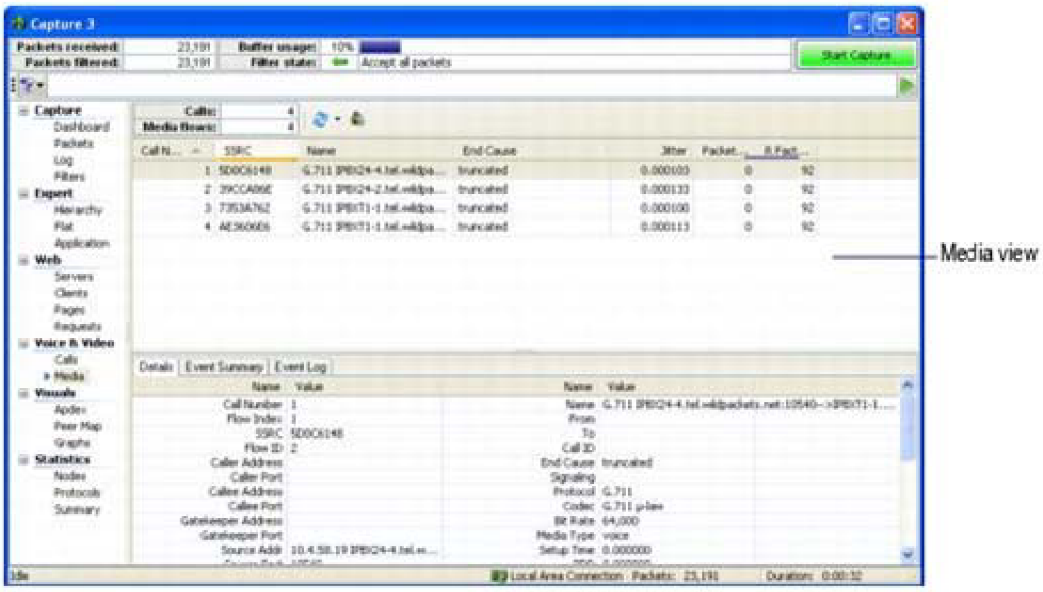


Figure 4.8: Voice and Video media view.

Additional information is provided in nested tabs for selected calls or media flows displayed in the upper pane of the Voice and Video view.

4.2.1.2.1 Voice and Video Details tab

In the Calls view Figure 4.9, the Details tab contains all the information about the call. Every column in the Calls view is displayed in the Details tab Figure 4.9:

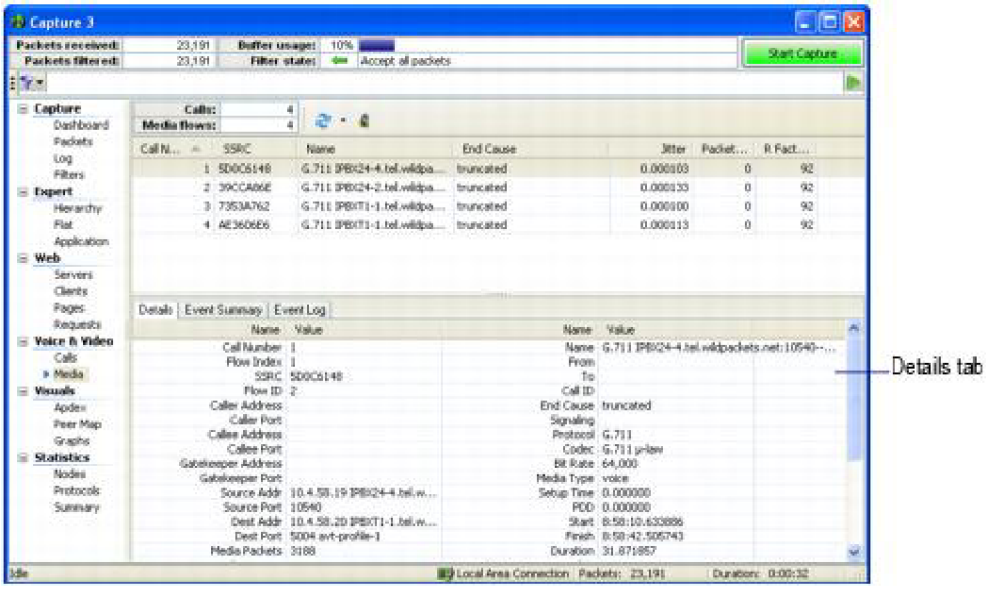


Figure 4.9: Details tab.

In the Media view, the Details tab displays details about the selected media flow and the call that contains it.

4.2.1.2.2 Voice and Video Event Summary tab

The Event Summary tab in Figure 4.10 shows a count of each voice and video expert event for this capture. Severity levels configured in the EventFinder are displayed to the left of each voice and video expert event. The Expert EventFinder contains many VoIP expert events, including those relating to H.225, MGCP, RTP, and SIP.

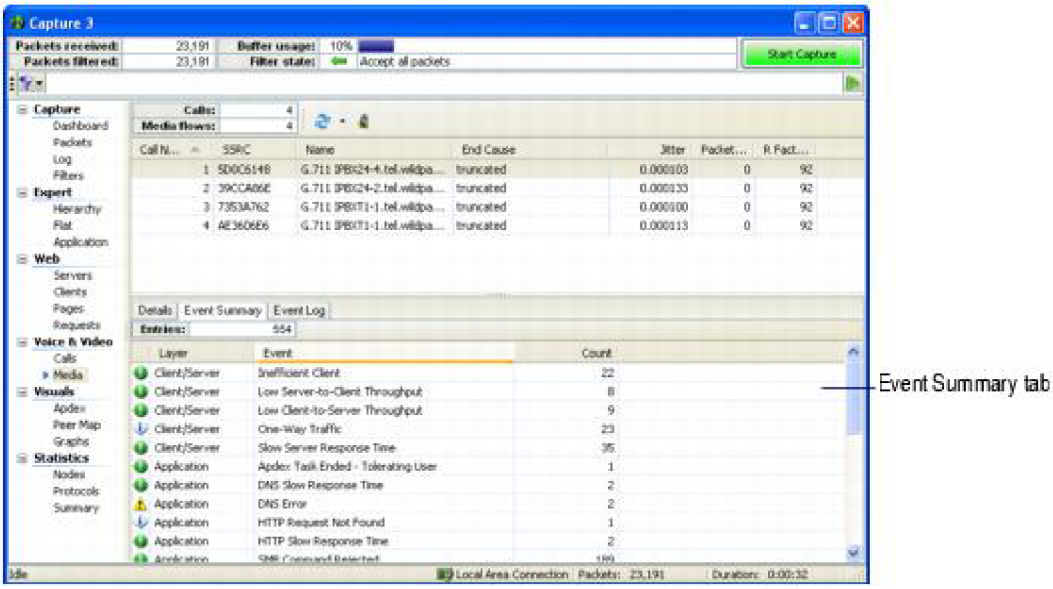


Figure 4.10: The Event Summary tab.

4.2.1.2.3 Voice and Video Event Log tab

The Event Log tab in Figure 4.11 shows a list of all voice and video expert events found in this capture. The four toggle buttons in the Event Log tab header let us have the ability to show or hide events by levels of severity.

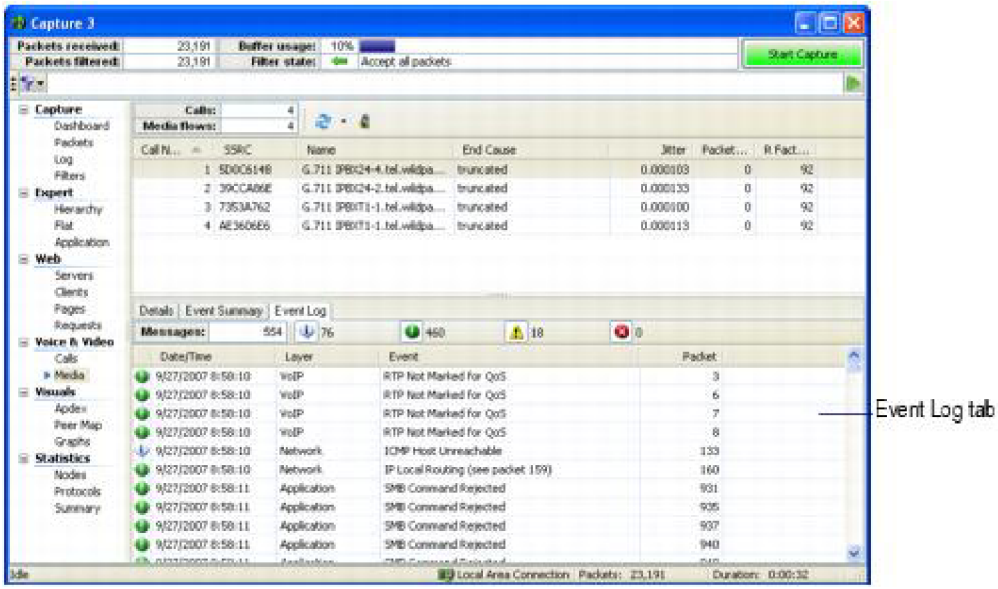


Figure 4.11: The Event Log tab.

4.2.2 Voice and Video Visual Expert

The Voice and Video Visual Expert in Figure 4.12 displays each individual packet of an entire call within a single window, as well as the RTP packet timing, jitter, and quality score over time. If there are gaps of missing or late RTP packets, these gaps are also displayed, along with their effect on call quality.

The Signaling tab of the Voice and Video Visual Expert window displays signal bounce diagrams with columns corresponding to each node participating in the call. Signaling and media stream packets are represented by horizontal lines, giving us an immediate overview of the contents of a call. The bounce diagram also includes linear representations as well as numerical measurements of R-Factor and jitter values.

In addition to many of the columns available in the Calls and Media views, the Voice and Video Visual Expert columns allow us to calculate the relative time lapse between individual packets, the signaling sequence method of the call, and more.

The parts of the Signaling tab are described below.

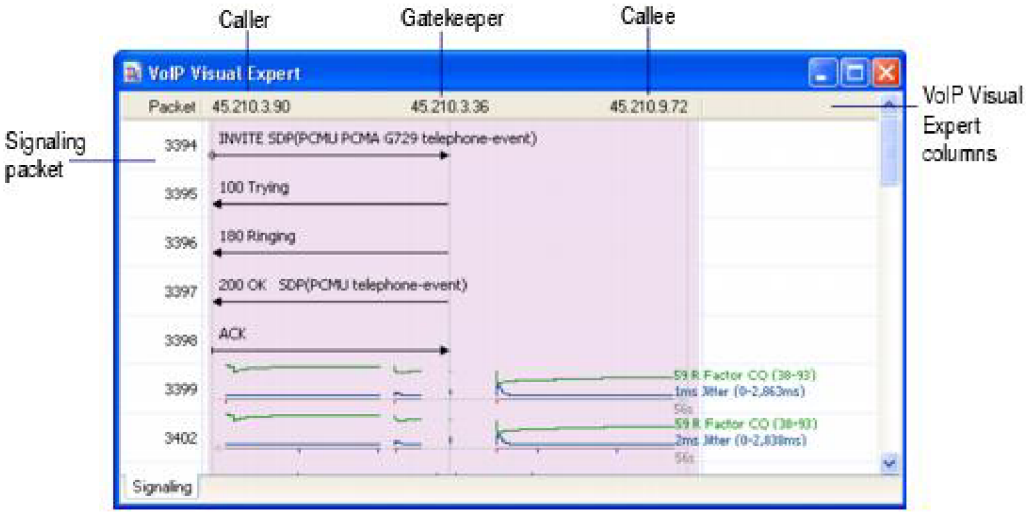
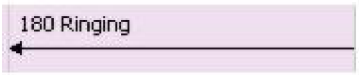


Figure 4.12: The Voice and Video Visual Expert.

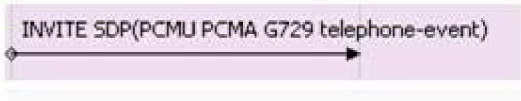
Each node participating in the call gets a vertical line, with the caller usually on the left, the gatekeeper in the middle, and callee on the right.

The Signaling packets:

- Each signaling packet appears as a black horizontal arrow, with a summary above the arrow:



- Packets that start a call (such as SIP INVITE packets) start with a small diamond:



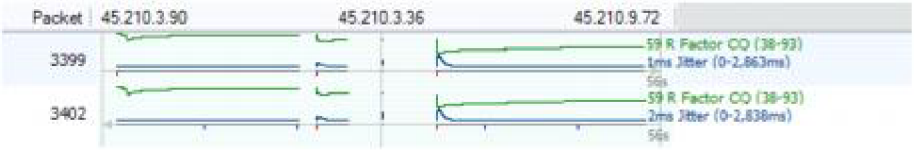
- Packets that usually mean the end of call setup (such as SIP ACK packets) start with a small bar. The time between these two packets is the call setup time.

RTP/RTCP packets: RTP/RTCP media packets appear as horizontal light grey arrows, with a green R-Factor and blue jitter line graph above the arrow showen in Figure 4.13.

Figure 4.13: RTP/RTCP media packets.

4.2.2 .1 RTP/RTCP Rows

The media or voice streams (RTP/RTCP packets) within a call display in the Signaling tab as rows progressing through time, with the first packet in the row at the left to the last packet at the right. Since most calls are bidirectional, a pair of rows often appears with one row for each direction. The parts of the RTP/RTCP media packets in a bidirectional call are identified below.



Grey arrows and numbers: Grey horizontal arrows represent the RTP/RTCP media packets. The last packet in the row displays a small grey number showing the entire duration for the row. (Trivial durations are not shown for very brief rows.)

Green lines and numbers: Green horizontal lines show R-Factor conversational values, with the row's final value and minimum-maximum range in green to the right of the last packet in the row.

Blue lines and numbers: Blue lines show jitter values, with the row's final value and minimum-maximum range in blue to the right of the last packet in the row.

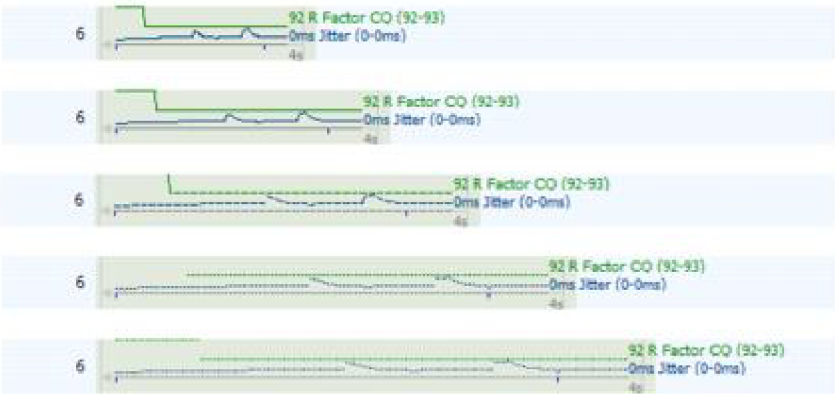
Blue tick marks: Blue tick marks represent RTCP packets.

Grey tick marks: Grey tick marks represent out-of-sequence RTP packets.

Red tick mark: Red tick marks show gaps of one or more missing packets.

Note: Gaps where no packets appear are readily visible, as well as their immediate effects of lowering R-Factor and raising jitter values.

As we expand the bounce diagram column, the Voice and Video Visual Expert can break an RTP line into its individual packets, as shown below:



4.2.3 Making a voice or video filter

For calls, we can create an address filter between caller and callee, caller and gateway, and gateway and callee. If these are three separate nodes, an advanced filter with three bidirectional address filters will be created, as shown in the Figure 4.14.

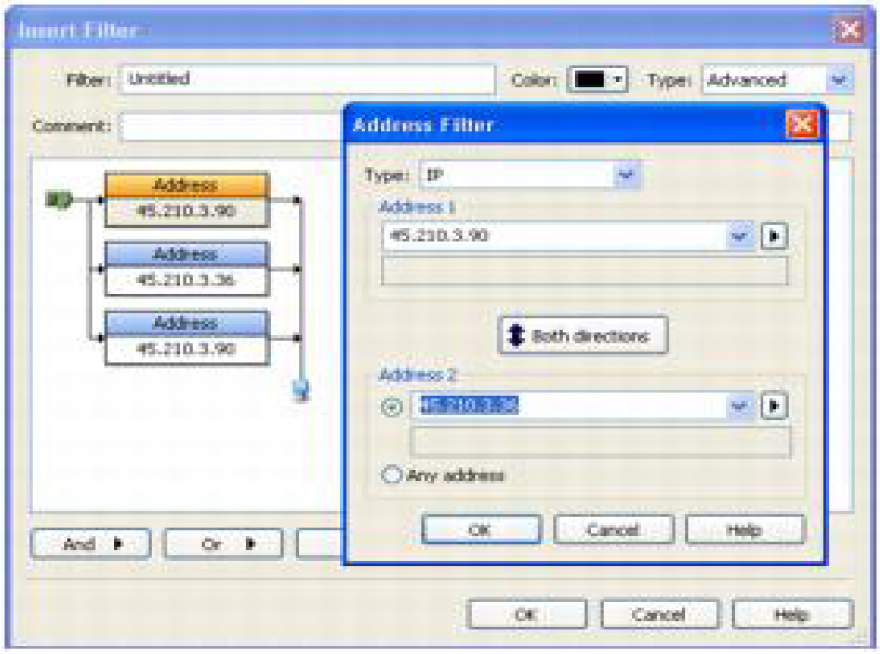


Figure 4.14: The address filter.

5. Laboratory exercise

5.1 Configuration

We are going to set a ADSL connection as a task for this lab exercise, the devices which we are going to use are an IP DSLAM, router, switch and multi- PVCs ADSL modem .The Figure 5.1 below illustrates how the connection is set.

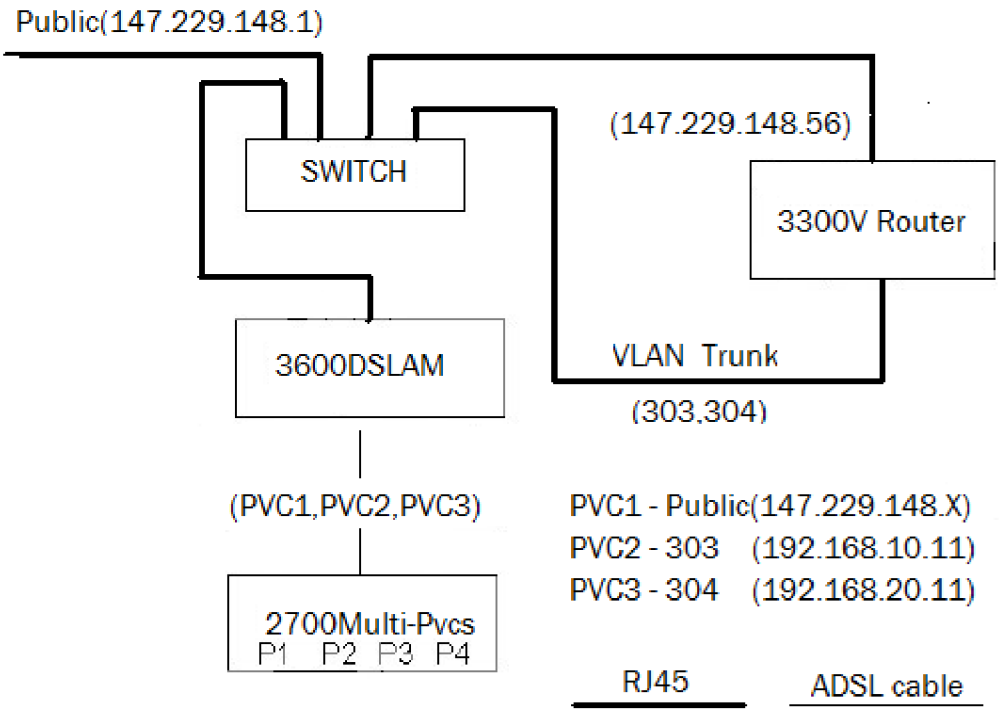


Figure 5.1 ADSL connection.

At first we created two VLAN's 303 for VOIP with IP address (192.168.10.11) port number 2, 304 IPTV with IP address (192.168.20.11) and port number 1 an 4 are for the public connection Through the (vigor 2700VG) in Figure 5.1 ,we created 3 channels and we had setup the VPI (Virtual Path Identifier) and VCI (Virtual channel identifier) that they are unique in each channel, we use the UBR as a Qos, only that it contains less capability than the other service's on ATM, so there is a great probability of losing the connection. We have chose MPoA which is an efficient solution to the interoperation between ATM and the existing network. And furthermore we have configured the type encapsulation: 1483 bridged IP LLC ,in order to make the modem work as a bridge than to route the data. Therefore we

untagged the data in the port which belongs from the router as it described in Figure 5.2.

Internet Access >> Multi-PVCs

Multi-PVCs

General		ATM QoS		Port-based Bridge		
Channel	Enable	VPI	VCI	QoS Type	Protocol	Encapsulation
1.	☑	8	48	UBR	MPoA	1483 Bridged IP LLC
2.	☐	8	35	UBR	MPoA	1483 Bridged IP LLC
3.	☐	8	36	UBR	MPoA	1483 Bridged IP LLC
4.	☑	8	35	UBR	MPoA	1483 Bridged IP LLC
5.	☑	8	36	UBR	MPoA	1483 Bridged IP LLC
6.	☐	8	53	UBR	PPPoE	LLC/SNAP
7.	☐	8	54	UBR	PPPoE	LLC/SNAP
8.	☐	8	55	UBR	PPPoE	LLC/SNAP

Note:VPI/VCI must be unique for each channel!

Figure 5.1: The configuration at the ADSL modem.

Advanced - LAN VLAN Setting

○ Disable ○ Port Base VLAN ◉ 802.1Q VLAN

Port Base VLAN | **802.1Q VLAN**

Group

Index	Active	Name	VLAN ID	Member P1	Member P2	Member P3	Member P4	Frame Tag Operation P1	Frame Tag Operation P2	Frame Tag Operation P3	Frame Tag Operation P4
1	☑	VoIP	303	☑	☑	☑	☐	Tagged	Untagged	Tagged	Tagged
2	☑	IPTV	304	☑	☑	☑	☐	Tagged	Tagged	Untagged	Tagged
3	☑	Pokusny	305	☑	☑	☑	☐	Tagged	Tagged	Tagged	Tagged
4	☑	VLAN8	4000	☐	☐	☐	☑	Tagged	Tagged	Tagged	Untagged

☑ Enable management port for P4

Port Setting

	P1	P2	P3	P4
Port VLAN ID	5	303	304	4000

Figure 5.2: The Router side from tagging and untagging.

We identify the ports in the modem at the (port-based bridge) as it is shown in Figure 5.3. We setup port number 2 for the VLAN 303, and number 3 for the VLAN 304.

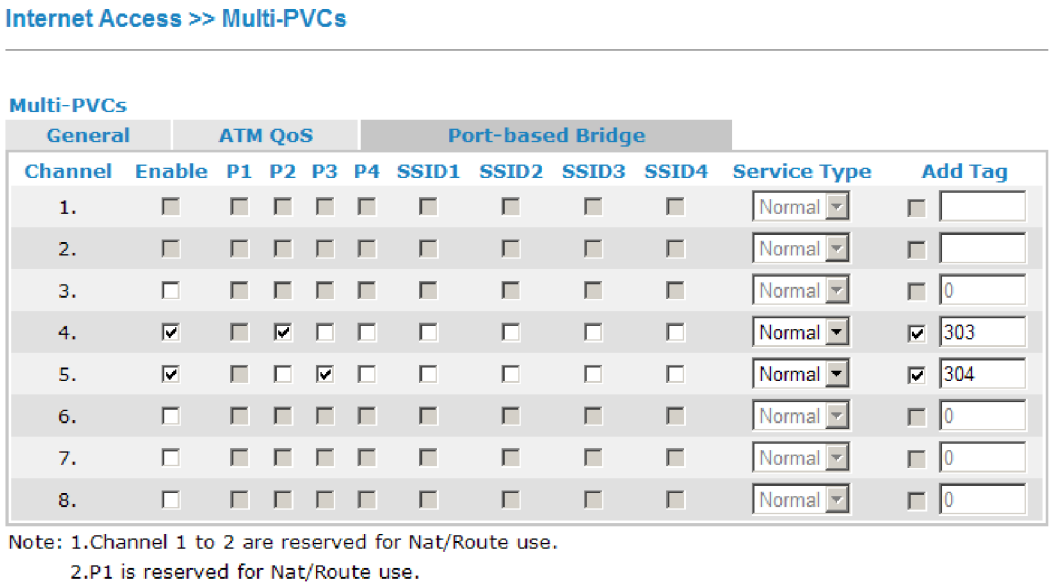


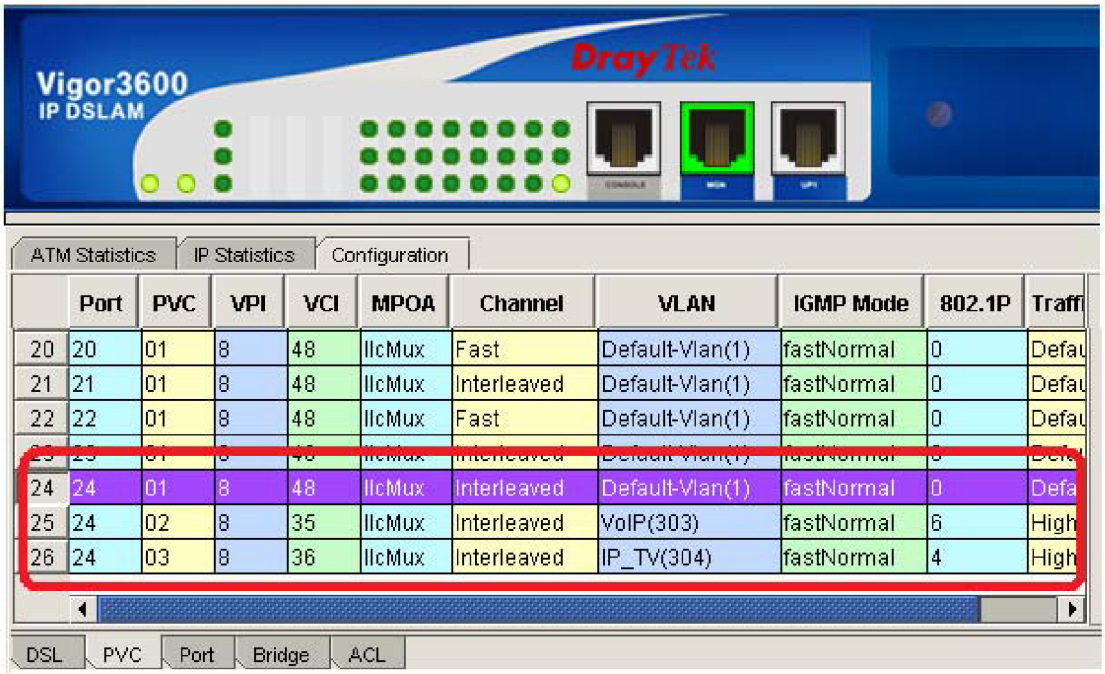
Figure 5.3: The port-based bridge at the ADSL modem.

The values of VPI and VCI which we created in CPE, is mapped to the IPDSLAM side Figure 5.4and Figure 5.5.

As a matter of fact, if we would like to use Video application, when Video server gets the video request from user client, it would send out video traffics to IPDSLAM with VLAN tag 304. IPDSLAM would parse the traffics as for video application and send traffics to video client.

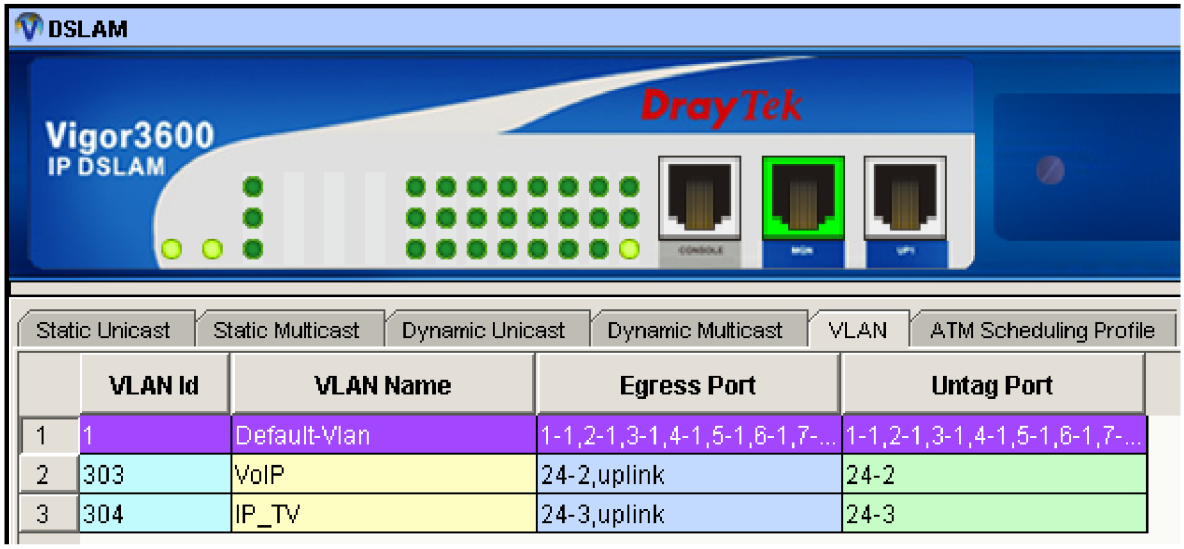
The other two applications are the same as mentioned above. However, the most important point is priority function within these three applications. If we run Video, VoIP and Internet at the same time, these three kinds of traffics would follow priority to ensure the best quality and reliability for special application like Video and VoIP.

If the priority of Video is set as “Highest”, this kind of traffics would get the first priority to be sent on a single high-capacity digital circuit at the same time.



	Port	PVC	VPI	VCI	MPOA	Channel	VLAN	IGMP Mode	802.1P	Traff
20	20	01	8	48	llcMux	Fast	Default-Vlan(1)	fastNormal	0	Defau
21	21	01	8	48	llcMux	Interleaved	Default-Vlan(1)	fastNormal	0	Defau
22	22	01	8	48	llcMux	Fast	Default-Vlan(1)	fastNormal	0	Defau
23	23	01	8	48	llcMux	Interleaved	Default-Vlan(1)	fastNormal	0	Defa
24	24	01	8	48	llcMux	Interleaved	Default-Vlan(1)	fastNormal	0	Defa
25	24	02	8	35	llcMux	Interleaved	VoIP(303)	fastNormal	6	High
26	24	03	8	36	llcMux	Interleaved	IP_TV(304)	fastNormal	4	High

Figure 5.4 : the VLAN's and the PVC's groups are mapped on the IPDISLAM at port 24.



	VLAN Id	VLAN Name	Egress Port	Untag Port
1	1	Default-Vlan	1-1,2-1,3-1,4-1,5-1,6-1,7-...	1-1,2-1,3-1,4-1,5-1,6-1,7-...
2	303	VoIP	24-2,uplink	24-2
3	304	IP_TV	24-3,uplink	24-3

Figure 5.5: another view of the DSLAM.

Here we have the information about the system and the online status shown in Figures 5.6 and 5.7.

System Status

Model Name	: Vigor2700 series
Firmware Version	: 2.8.1_131812
Build Date/Time	: May 30 2008 14:10:55
ADSL Firmware Version	: 131812_B Annex B

LAN

MAC Address	: 00-50-7F-DE-BB-E0	
1st IP Address	: 192.168.1.1	
1st Subnet Mask	: 255.255.255.0	
DHCP Server	: Yes	

VoIP

Port	: 1	2
SIP registrar	:	
Account ID	: change_me	change_me
Register	:	
Codec	:	
In Calls	: 0	0
Out Calls	: 0	0

WAN

Link Status	: Disconnected
MAC Address	: 00-50-7F-DE-BB-E1
Connection	: ---
IP Address	: ---
Default Gateway	: ---
DNS	: 194.109.6.66

Wireless LAN

MAC Address	: 00-50-7f-de-bb-e0
Frequency Domain	: Europe
Firmware Version	: 1.0.8.0

Figure 5.6: the system status on the ADSL modem.

Online Status

System Status — **System Uptime: 0:28:33**

LAN Status — **Primary DNS:** 194.109.6.66 — **Secondary DNS:** 194.98.0.1

IP Address	TX Packets	RX Packets
192.168.1.1	11420	8750

WAN Status — **GW IP Addr:** ---

Mode	IP Address	TX Packets	TX Rate	RX Packets	RX Rate	Up Time
---	---	0	0	0	0	00:00:00

ADSL Information (ADSL Firmware Version: 131812_B)

ATM Statistics	TX Blocks	RX Blocks	Corrected Blocks	Uncorrected Blocks
	47301	54749	0	1

ADSL Status	Mode	State	Up Speed	Down Speed	SNR Margin	Loop Att.
	ADSL2+ (G.992.5)	SHOWTIME	1344000	7668000	30	0

Figure 5.7: the online status on the ADSL modem.

The Figure 5.8 below at the DSLAM shows the upstream and the downstream values. Here is where we can increase and decrease the bit rate of the multiplexer (DSLAM).

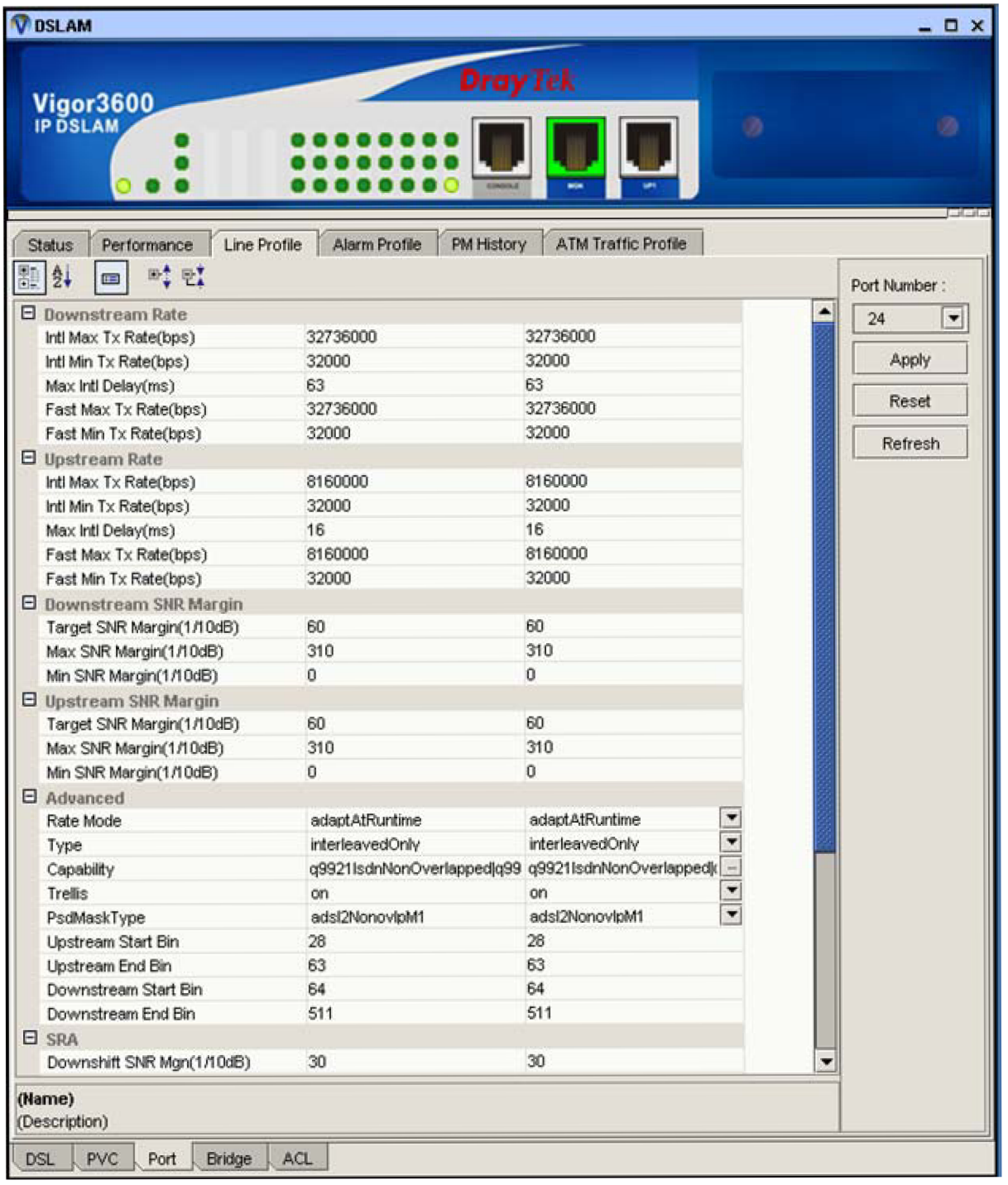


Figure 5.8: The upstream and downstream values in DSLAM

5.2 Laboratory Results

We divided our **measurement** into several steps:

1. PUBLIC connection with and without load at port 1, with High bit rate of Multiplexer.
2. VLAN connection with and without load at port 2, with High bit rate of Multiplexer
3. VLAN connection with and without load at port 2, with Low bit rate of Multiplexer.
4. PUBLIC connection with and without load at port 1, with Low bit rate of Multiplexer.
5. VLAN connection with and without load at port 2, with various bit rate of multiplexer.
6. PUBLIC connection from far distance, with High bit rate Multiplexer.

5.2.1.1 Public connection without load (high bit rate)

At first we connect the IP phone and the PC in port number 1, through a hub. We observed the monitor displaying the result from the two softwares which we were using.

From the wireshark and the OmniPeek we obtained the following table 5.9, two streams Figure 5.10 and the jitter graphs for the uplink and downlink are in Figure 5.11 and Figure 5.12:

7.6 Mbps	PUBLIC High bit-rate			
	Chanel Released		Chanel Overloaded	
	upstream	downstream	Upstream	Downstream
Total packets	6019	6001	9225	9212
packets lost	0	1	0	10
MAX delta (ms)	22.13	72.27	81.07	147.55
MAX Jitter (ms)	2.12	15.8	7	21.54
Mean jitter (ms)	1.74	3.55	0.58	14.75
R-Factor	92	90	92	88
CQ	(92-92)	(89-92)	(92-92)	(86-92)

Table 5.9 Public connection high bit rate.

From the wireshark we obtained the following two streams and the graphs:

Detected 2 RTP streams. Choose one for forward and reverse direction for analysis

Src IP addr	Src port	Dest IP addr	Dest port	SSRC	Payload	Packets	Lost	Max Delta (ms)	Max Jitter (ms)	Mean Jitter (ms)	Pb?
147.229.148.111	16412	147.229.151.125	10262	0x2251767E	ITU-T G.711 PCMA	6019	0 (0.0%)	22.13	2.12	1.74	
147.229.151.125	10262	147.229.148.111	16412	0x4C66C96A	ITU-T G.711 PCMA	6001	1 (0.0%)	72.27	15.80	3.55	X

Forward: 147.229.148.111:16412 -> 147.229.151.125:10262, SSRC=0x2251767E

Figure 5.10: RTP streams for Public connection without load (high bit rate).

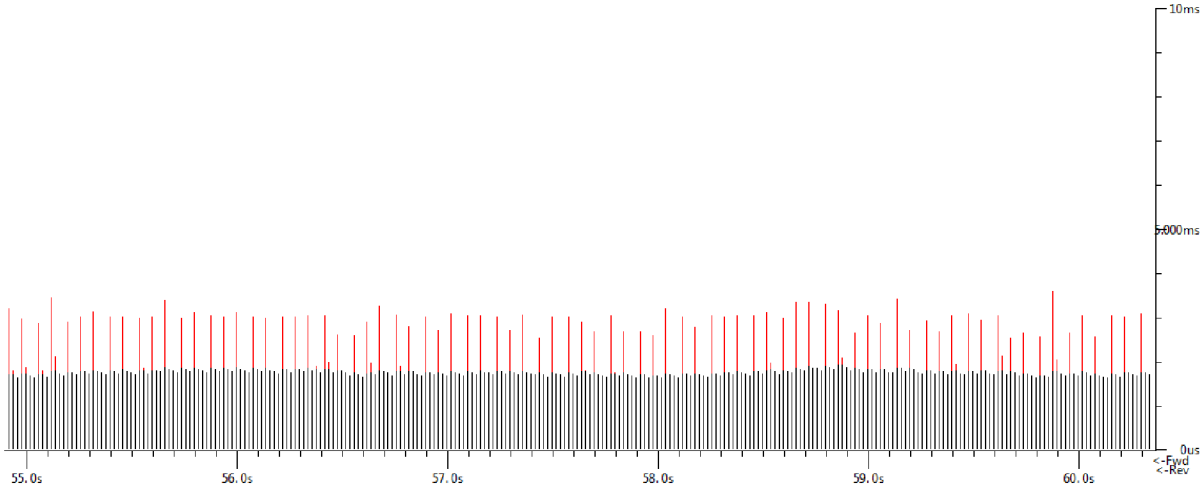


Figure 5.11: the uplink graph.

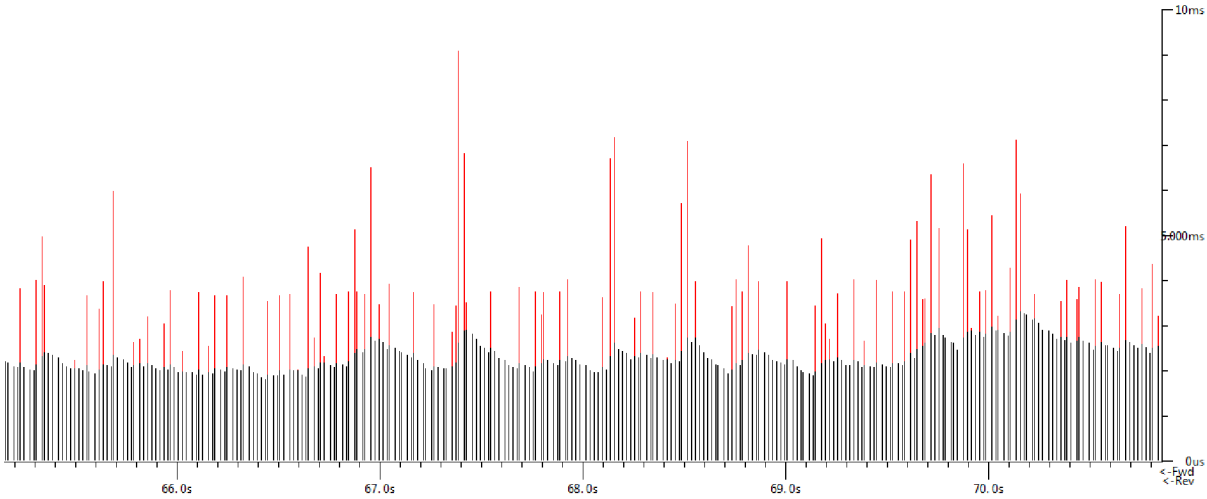


Figure 5.12: The downlink graph.

Below in Figure 5.13 is the result obtained by the omnipeak:

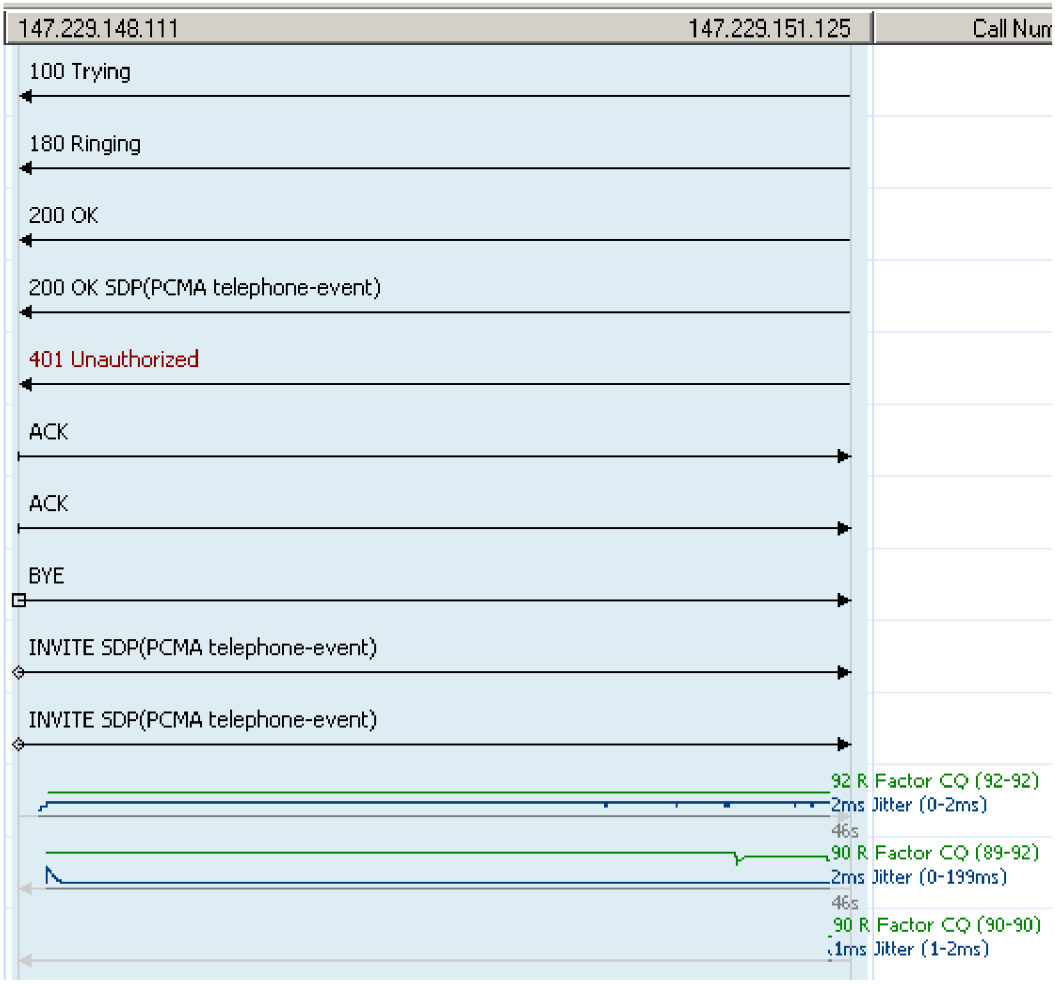


Figure 5.13: The view of R-Factors for both streams from omnipeak.

5.2.1.2 Public connection with load (high bit rate)

We connected another PC to port number 4 on the CPE, followed by downloading the data. From the Figure 5.14 below we can see the usage of the multiplexer (DSLAM) is up to 5.18Mbit/s, we started the programs prior to making the call.

In the previous Table 5.9 We merged the result of the uploading and without uploading to clarify the comparison between them. The obtained two streams are in Figure 5.15 and the jitter graphs for the uplink and downlink are in Figures 5.16 and Figure 5.17:

Status | Summary | Throughput | Instant Rate | DHCP Opt82 | Subscriber

	Port	PVC	Rx(Kbps)	Tx(Kbps)	Rx(pkt/s)	Tx(pkt/s)
1	1	1	0.0	0.0	0.0	0.0
2	2	1	0.0	0.0	0.0	0.0
3	3	1	0.0	0.0	0.0	0.0
4	4	1	0.0	0.0	0.0	0.0
5	5	1	0.0	0.0	0.0	0.0
6	6	1	0.0	0.0	0.0	0.0
7	7	1	0.0	0.0	0.0	0.0
8	8	1	0.0	0.0	0.0	0.0
9	9	1	0.0	0.0	0.0	0.0
10	10	1	0.0	0.0	0.0	0.0
11	11	1	0.0	0.0	0.0	0.0
12	12	1	0.0	0.0	0.0	0.0
13	13	1	0.0	0.0	0.0	0.0
14	14	1	0.0	0.0	0.0	0.0
15	15	1	0.0	0.0	0.0	0.0
16	16	1	0.0	0.0	0.0	0.0
17	17	1	0.0	0.0	0.0	0.0
18	18	1	0.0	0.0	0.0	0.0
19	19	1	0.0	0.0	0.0	0.0
20	20	1	0.0	0.0	0.0	0.0
21	21	1	0.0	0.0	0.0	0.0
22	22	1	0.0	0.0	0.0	0.0
23	23	1	0.0	0.0	0.0	0.0
24	24	1	95.83	4144.6	196.42	353.5
25	24	2	31.78	481.59	31.45	49.01
26	24	3	0.0	0.0	0.0	0.0

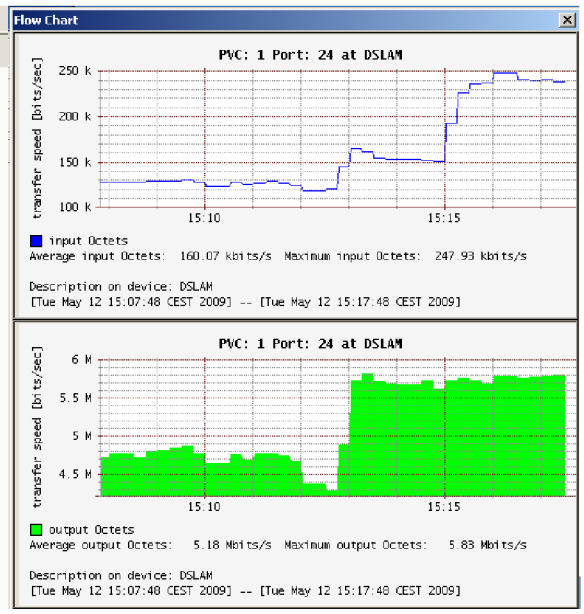


Figure 5.14: Data flow from the DSLAM with average of 5.18Mbit/s.

Detected 2 RTP streams. Choose one for forward and reverse direction for analysis

Src IP addr	Src port	Dest IP addr	Dest port	SSRC	Payload	Packets	Lost	Max Delta (ms)	Max Jitter (ms)	Mean Jitter (ms)	Pb?
147.229.148.111	16454	147.229.151.125	15220	0xD4EDB796	ITU-T G.711 PCMA	9225	0 (0.0%)	81.07	7.00	0.58	
147.229.151.125	15220	147.229.148.111	16454	0xBAB6D7B3	ITU-T G.711 PCMA	9212	10 (0.1%)	147.55	21.54	14.75	X

Forward: 147.229.148.111:16454 -> 147.229.151.125:15220, SSRC=0xD4EDB796

Figure 5.15: The RTP streams of Public connection with load (high bit rate).

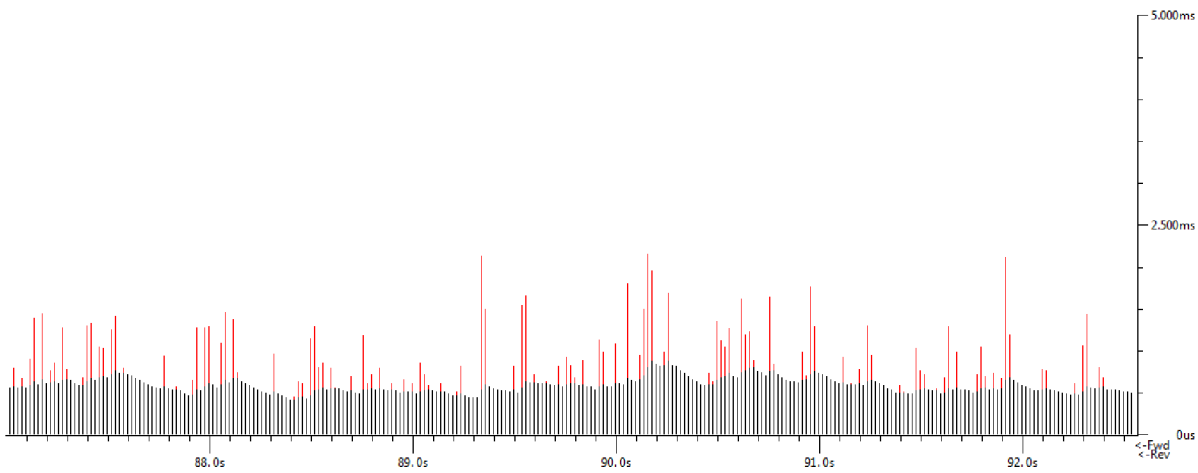


Figure: 5.16: the uplink graph.

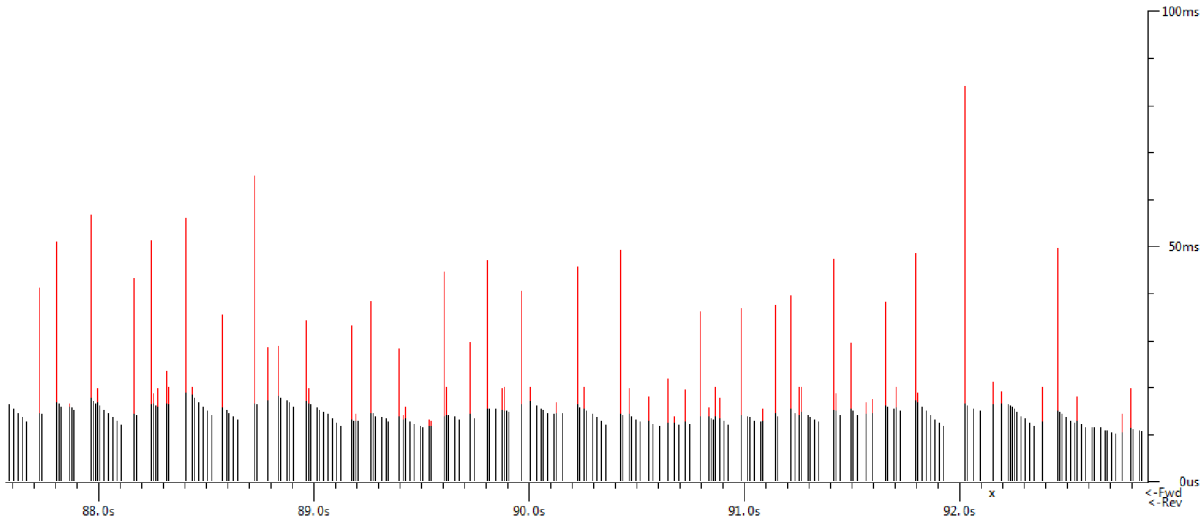


Figure 5.17: The downlink graph.

Below in Figure 5.18 is the result obtained by the omnipeak:

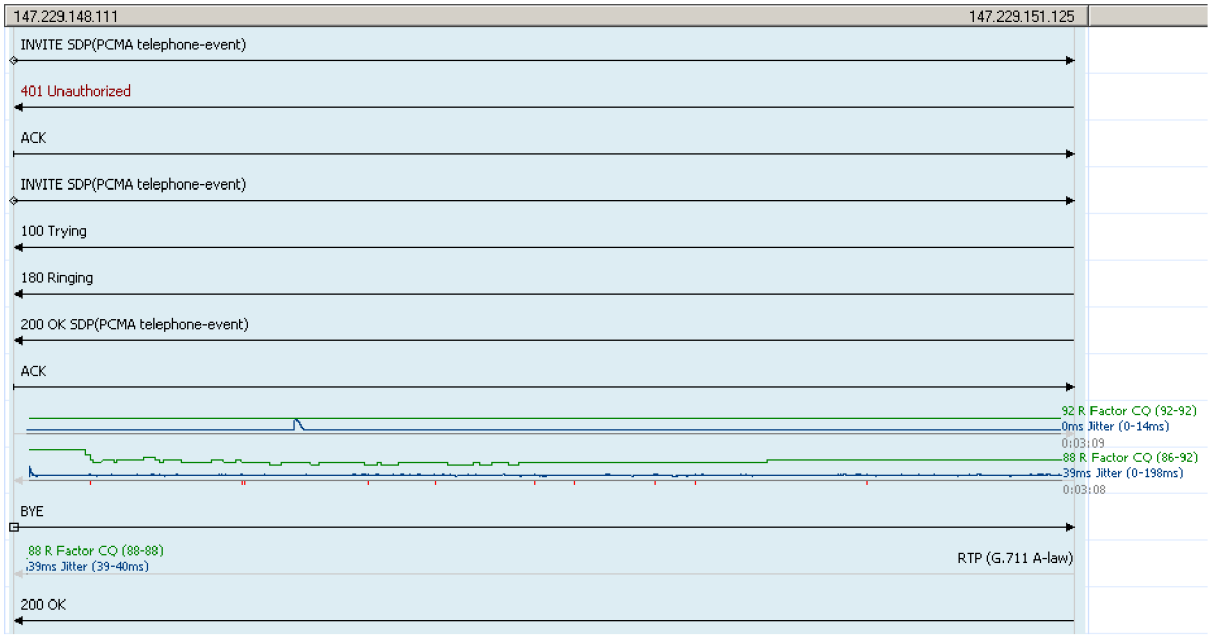


Figure 5.18: The view of R-Factors for both streams from omnipeak.

5.2.2.1 VLAN Connection without load (high bit rate)

We shall proceed as the previous step, but we removed the wire from port number 1 and reconnected the wire to port number 2 where we create the 303 VLAN

connection, we tested the connection with High bit-Rate without receiving any data, as we can see below in table 5.19 which shows the values obtained.

The table displays the obtained results from the omnipeak and wireshark:

7.6 Mbps	VLAN 303 High bit-Rate			
	Chanel Released		Chanel Overloaded	
	Upstream	Downstream	Upstream	Downstream
Total packets	4670	4672	7912	7912
packets lost	0	0	0	0
MAX delta (ms)	26.09	31.27	74.86	92.44
MAX Jitter (ms)	1.24	15.52	6.35	15.62
Mean jitter (ms)	0.56	1.83	0.46	1.82
R-Factor	92	89	92	89
CQ	(92-92)	(89-92)	(92-92)	(89-92)

Table 5.19: VLAN 303 connection (High bit-Rate).

As shown below in figure 5.20 we can see the usage of the DSLAM, the graph shows the bit rate of the connection. The status of the DSLAM does not show any data flow. The obtained two streams are in Figure 5.21 and the jitter graphs for the uplink and downlink are in Figures 5.22 and Figure 5.23:

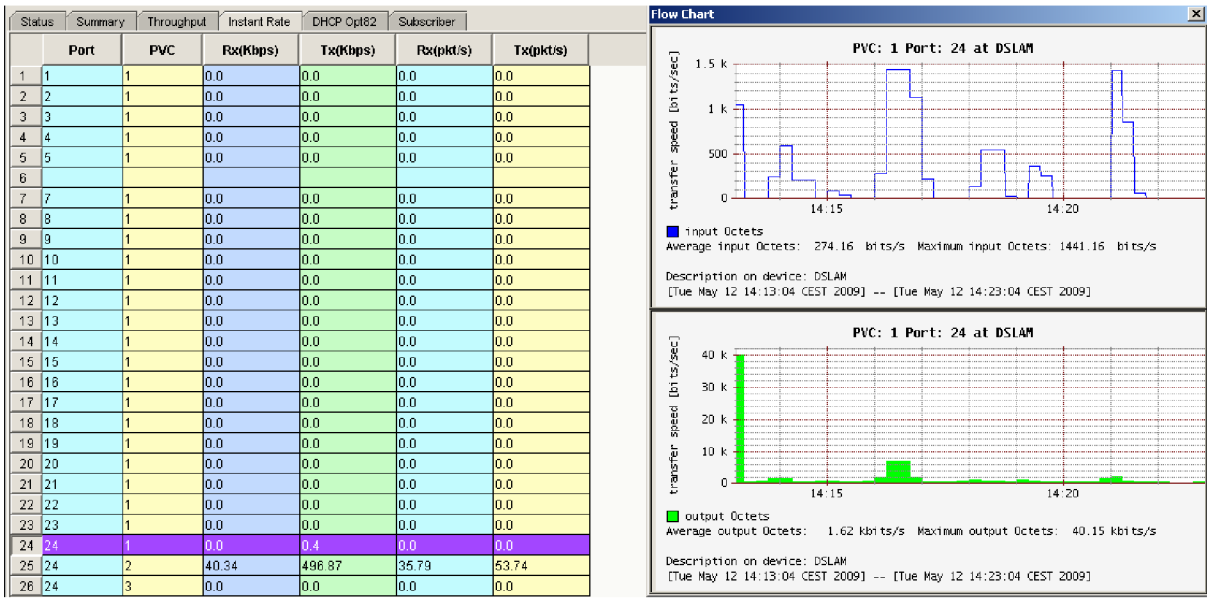


Figure 5.20: the usage of the DSLAM in port number 24 shows the range between 1.26 Kbps.

Detected 2 RTP streams. Choose one for forward and reverse direction for analysis

Src IP addr	Src port	Dest IP addr	Dest port	SSRC	Payload	Packets	Lost	Max Delta (ms)	Max Jitter (ms)	Mean Jitter (ms)	Pb?
192.168.10.109	16440	192.168.10.125	17660	0x2A8B2EA0	ITU-T G.711 PCMA	4670	0 (0.0%)	26.09	1.24	0.56	
192.168.10.125	17660	192.168.10.109	16440	0x88AFC5E8	ITU-T G.711 PCMA	4672	0 (0.0%)	31.27	15.52	1.83	

Forward: 192.168.10.109:16440 -> 192.168.10.125:17660, SSRC=0x2A8B2EA0

Figure 5.21: The two RTP streams downlink and uplink VLAN connection (High bit-rate).

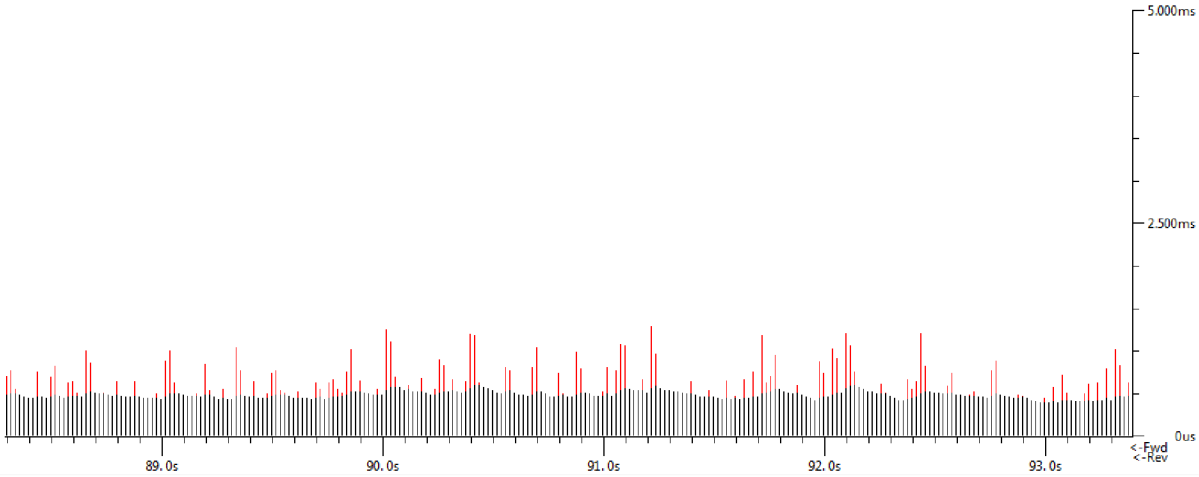


Figure 5.22: The uplink graph.

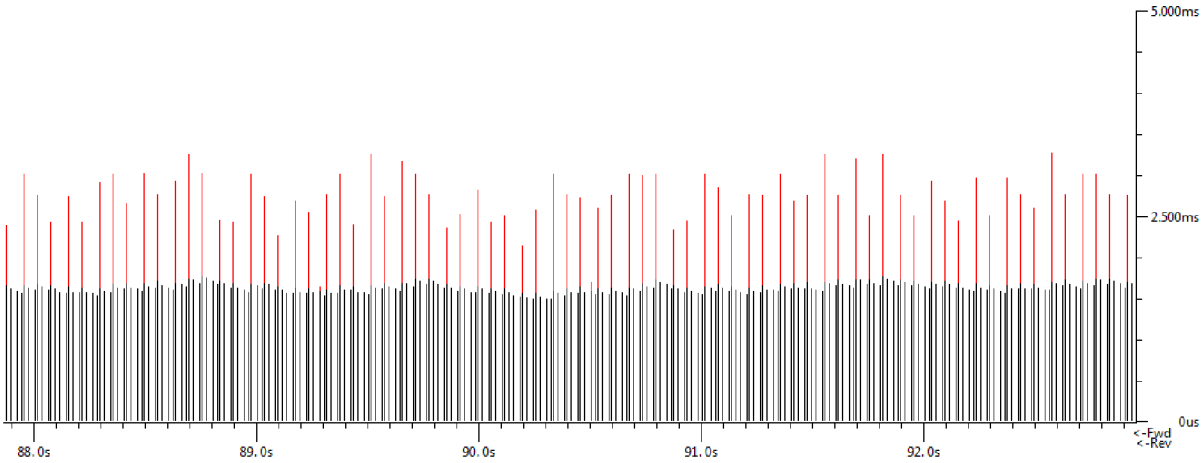


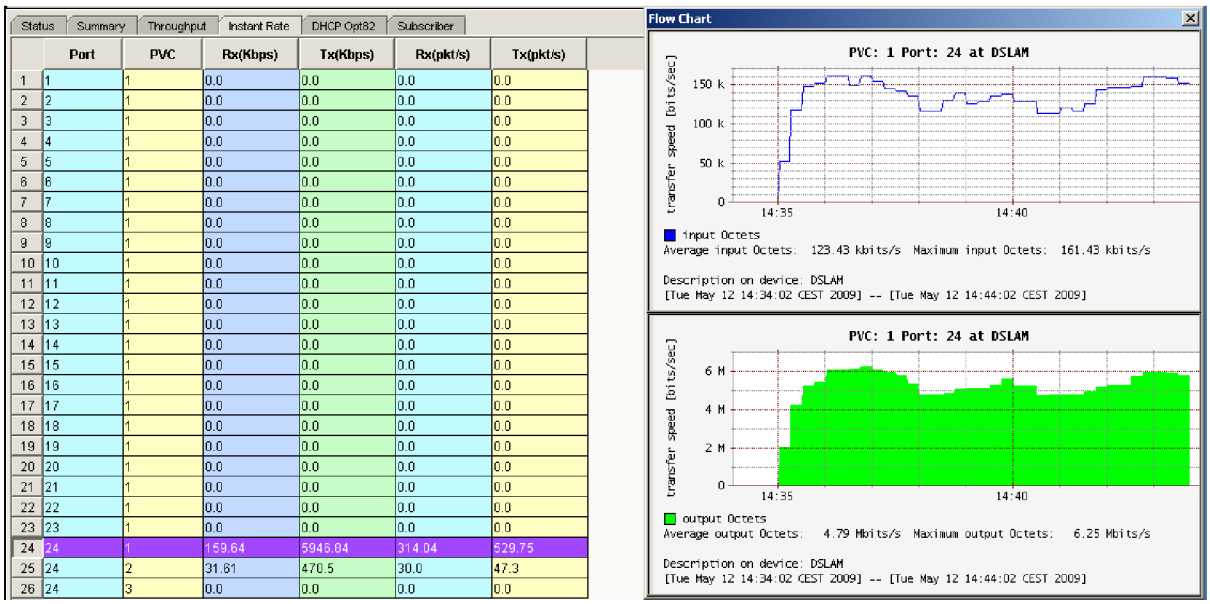
Figure 5.23: The downlink graph.

Below in Figure 5.24 is the result obtained by the omnipeak:

Figure 5.24: The view of R-Factors for both streams from omnipeak.

5.2.2.2 VLAN **Connection with load (high bit rate)**

We upload data from The PC which is connected to port number 4, as we can see below in figure 5.25 The usage of the DSLAM shows the flow average 4.79 Mbps. The result is merged to the previous Table 5.19. The obtained two streams are in Figure 5.26 and the jitter graphs are in Figures 5.27 and Figure 5.28:



Status | Summary | Throughput | Instant Rate | DHCP Opt82 | Subscriber

	Port	PVC	Rx(Kbps)	Tx(Kbps)	Rx(pkt/s)	Tx(pkt/s)
1	1	1	0.0	0.0	0.0	0.0
2	2	1	0.0	0.0	0.0	0.0
3	3	1	0.0	0.0	0.0	0.0
4	4	1	0.0	0.0	0.0	0.0
5	5	1	0.0	0.0	0.0	0.0
6	6	1	0.0	0.0	0.0	0.0
7	7	1	0.0	0.0	0.0	0.0
8	8	1	0.0	0.0	0.0	0.0
9	9	1	0.0	0.0	0.0	0.0
10	10	1	0.0	0.0	0.0	0.0
11	11	1	0.0	0.0	0.0	0.0
12	12	1	0.0	0.0	0.0	0.0
13	13	1	0.0	0.0	0.0	0.0
14	14	1	0.0	0.0	0.0	0.0
15	15	1	0.0	0.0	0.0	0.0
16	16	1	0.0	0.0	0.0	0.0
17	17	1	0.0	0.0	0.0	0.0
18	18	1	0.0	0.0	0.0	0.0
19	19	1	0.0	0.0	0.0	0.0
20	20	1	0.0	0.0	0.0	0.0
21	21	1	0.0	0.0	0.0	0.0
22	22	1	0.0	0.0	0.0	0.0
23	23	1	0.0	0.0	0.0	0.0
24	24	1	159.64	5946.84	314.04	529.75
25	24	2	31.81	470.5	30.0	47.3
26	24	3	0.0	0.0	0.0	0.0

Figure 5.25: The usage of the DSLAM.

Detected 2 RTP streams. Choose one for forward and reverse direction for analysis

Src IP addr	Src port	Dest IP addr	Dest port	SSRC	Payload	Packets	Lost	Max Delta (ms)	Max Jitter (ms)	Mean Jitter (ms)	Pb?
192.168.10.109	16470	192.168.10.125	17982	0xE695D995	ITU-T G.711 PCMA	7912	0 (0.0%)	74.86	6.35	0.46	
192.168.10.125	17982	192.168.10.109	16470	0xDB1576CA	ITU-T G.711 PCMA	7912	0 (0.0%)	92.44	15.62	1.82	

Forward: 192.168.10.109:16470 -> 192.168.10.125:17982, SSRC=0xE695D995

Figure 5.26: The two RTP streams for VLAN connection (High bit rate).

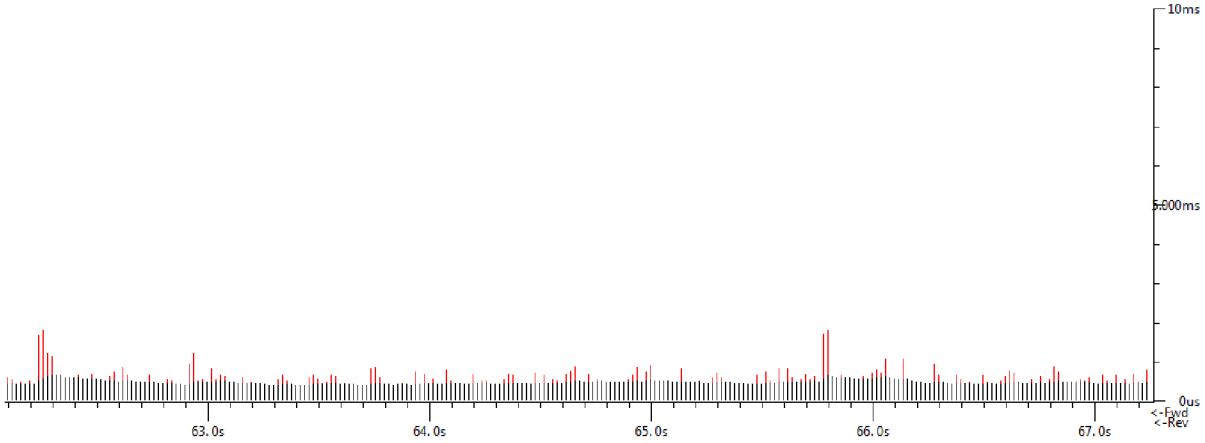


Figure 5.27: The uplink graph.

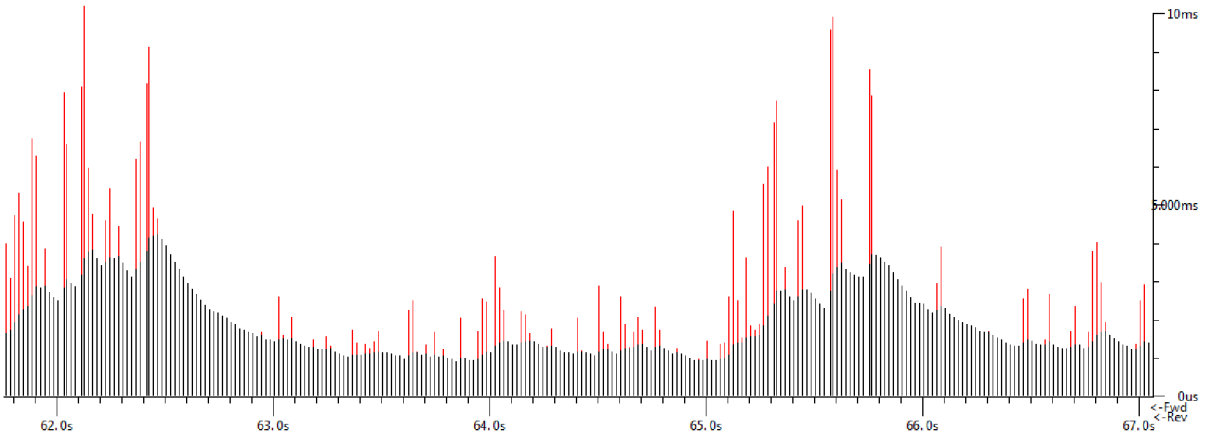


Figure 5.28: The downlink graph.

Below in Figure 5.29 is the result obtained by the omnipeak:

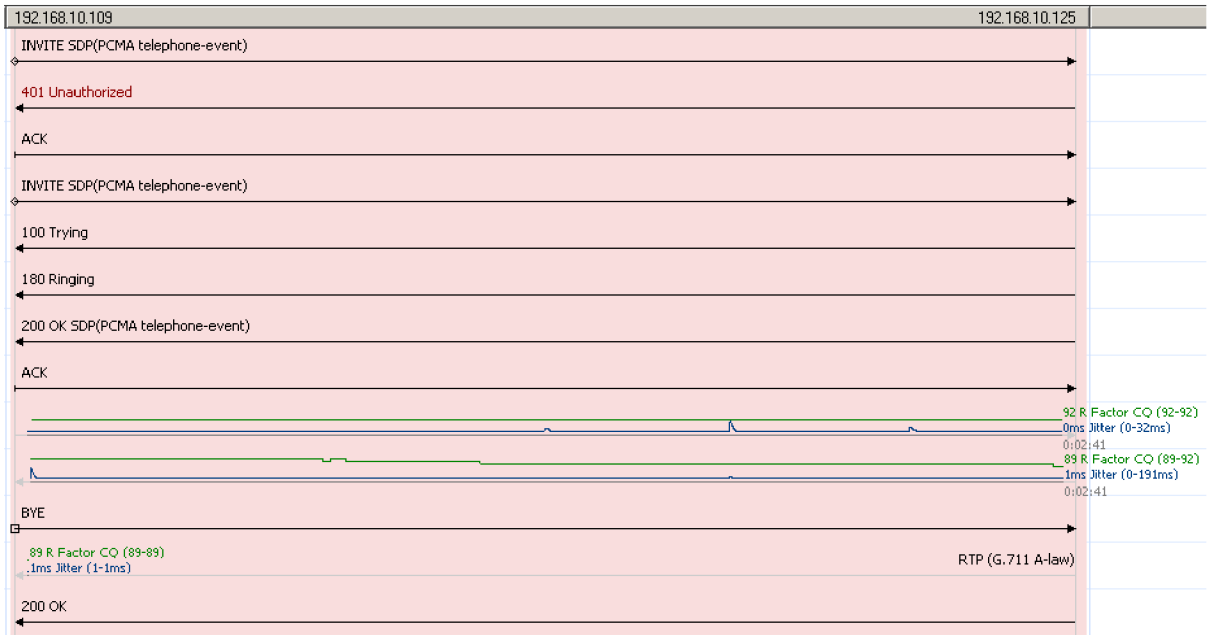


Figure 5.29: The view of R-Factors for both streams from omnipeak.

5.2.3.1 303 VLAN Connection without load (Low bandwidth)

We decreased the bit rate of the multiplexer without downloading any data, we made the call, and then obtained the values in Table 5.30.The obtained two streams are in Figure 5.32 and the jitter graphs are in Figures 5.33 and Figure 5.34:

300 Kbps	VLAN 303 Low bit-rate			
	Chanel Released		Chanel Overloaded	
	Upstream	Downstream	upstream	Downstream
Total packets	12368	12368	30124	30134
Packets lost	0	0	0	0
MAX delta (ms)	24.81	34.49	91.12	111.54
MAX Jitter (ms)	1.01	11.86	7.82	22.5
Mean jitter (ms)	0.43	0.99	0.62	16.77
R-Factor	92	91	92	90
CQ	(92-92)	(91-92)	(92-92)	(81-92)

Table 5.30: VLAN Connection Low bandwidth

Status | Summary | Throughput | Instant Rate | DHCP Opt82 | Subscriber

	Port	PVC	Rx(Kbps)	Tx(Kbps)	Rx(pkt/s)	Tx(pkt/s)
1	1	1	0.0	0.0	0.0	0.0
2	2	1	0.0	0.0	0.0	0.0
3	3	1	0.0	0.0	0.0	0.0
4	4	1	0.0	0.0	0.0	0.0
5	5	1	0.0	0.0	0.0	0.0
6	6	1	0.0	0.0	0.0	0.0
7	7	1	0.0	0.0	0.0	0.0
8	8	1	0.0	0.0	0.0	0.0
9	9	1	0.0	0.0	0.0	0.0
10	10	1	0.0	0.0	0.0	0.0
11	11	1	0.0	0.0	0.0	0.0
12	12	1	0.0	0.0	0.0	0.0
13	13	1	0.0	0.0	0.0	0.0
14	14	1	0.0	0.0	0.0	0.0
15	15	1	0.0	0.0	0.0	0.0
16	16	1	0.0	0.0	0.0	0.0
17	17	1	0.0	0.0	0.0	0.0
18	18	1	0.0	0.0	0.0	0.0
19	19	1	0.0	0.0	0.0	0.0
20	20	1	0.0	0.0	0.0	0.0
21	21	1	0.0	0.0	0.0	0.0
22	22	1	0.0	0.0	0.0	0.0
23	23	1	0.0	0.0	0.0	0.0
24	24	1	11.66	277.04	17.24	25.24
25	24	2	0.0	0.0	0.0	0.0
26	24	3	0.0	0.0	0.0	0.0

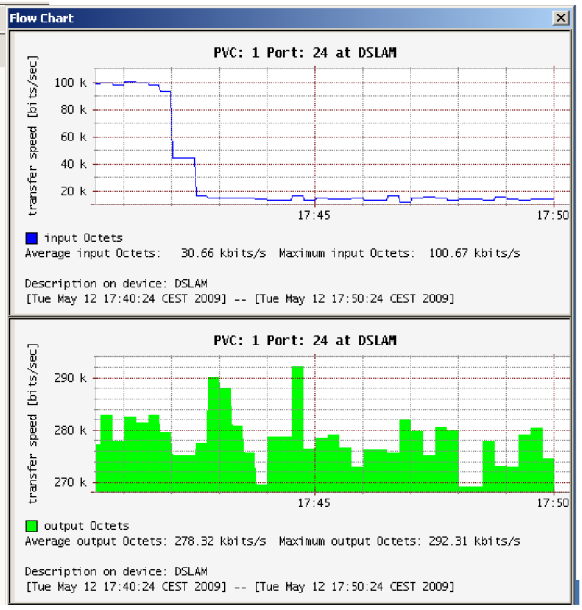


Figure 5.31: The usage of DSLAM.

Detected 2 RTP streams. Choose one for forward and reverse direction for analysis

Src IP addr	Src port	Dest IP addr	Dest port	SSRC	Payload	Packets	Lost	Max Delta (ms)	Max Jitter (ms)	Mean Jitter (ms)	Pb?
192.168.10.109	16446	192.168.10.125	12488	0xF1C817BB	ITU-T G.711 PCMA	12368	0 (0.0%)	24.81	1.01	0.43	
192.168.10.125	12488	192.168.10.109	16446	0x85A85BB4	ITU-T G.711 PCMA	12368	0 (0.0%)	34.49	11.86	0.99	

Forward: 192.168.10.109:16446 -> 192.168.10.125:12488, SSRC=0xF1C817BB

Figure 5.32: The two RTP streams for VLAN connection (Low bit rate)

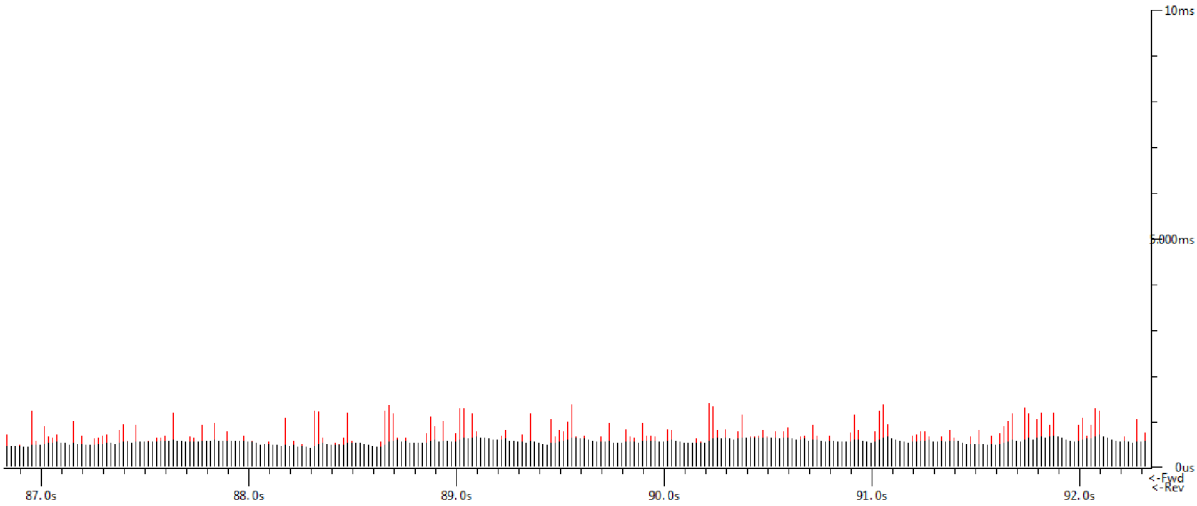


Figure 5.33: The uplink graph

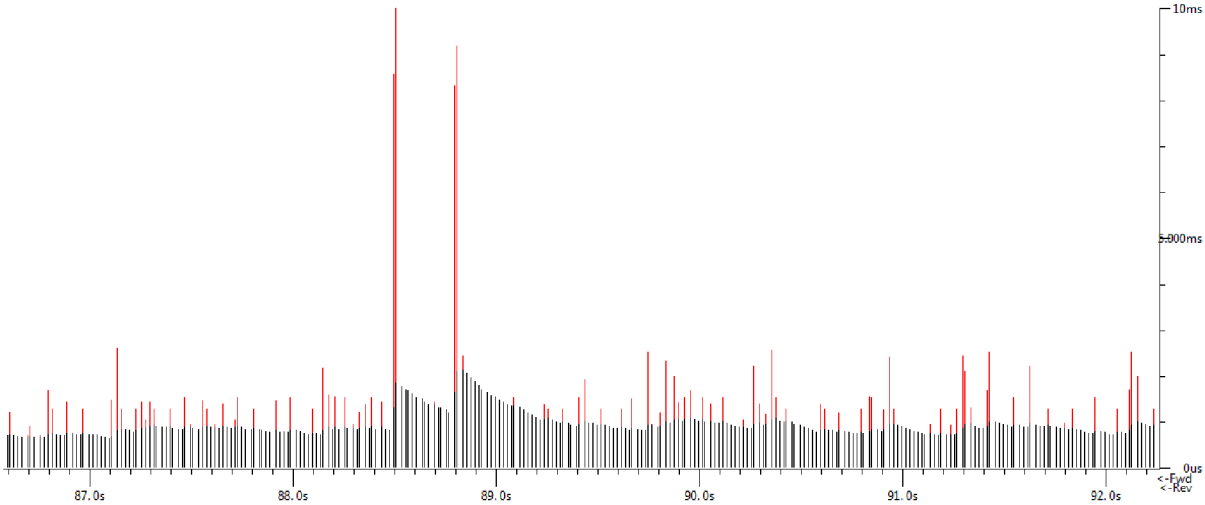


Figure 5.34: The downlink graph

Below in Figure 5.35 is the result obtained by the omnipeak:

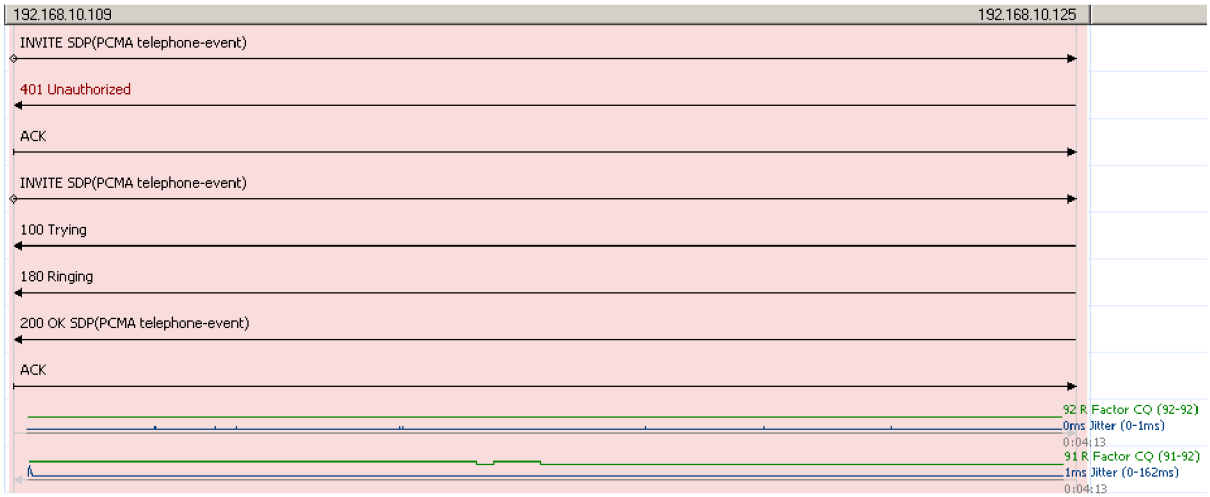


Figure 5.35: The view of R-Factors for both streams from omnipeak.

5.2.3.2 VLAN Connection with load (Low bandwidth)

In this case we downloaded the data and the values acquired that are displayed below in Figure 5.36, the jitter graphs are in Figures 5.33 and Figure 5.34 and then

we merged the results shown in the Table 5.30 in the previous section to compare the results.

Detected 2 RTP streams. Choose one for forward and reverse direction for analysis

Src IP addr	Src port	Dest IP addr	Dest port	SSRC	Payload	Packets	Lost	Max Delta (ms)	Max Jitter (ms)	Mean Jitter (ms)	Pb?
192.168.10.109	16448	192.168.10.125	13212	0x1B739AC1	ITU-T G.711 PCMA	30124	0 (0.0%)	91.12	7.82	0.62	
192.168.10.125	13212	192.168.10.109	16448	0x5E381F76	ITU-T G.711 PCMA	30134	0 (0.0%)	111.54	22.50	16.77	

Forward: 192.168.10.109:16448 -> 192.168.10.125:13212, SSRC=0x1B739AC1

Figure 5.36: The two RTP streams for VLAN connection (Low bit rate)

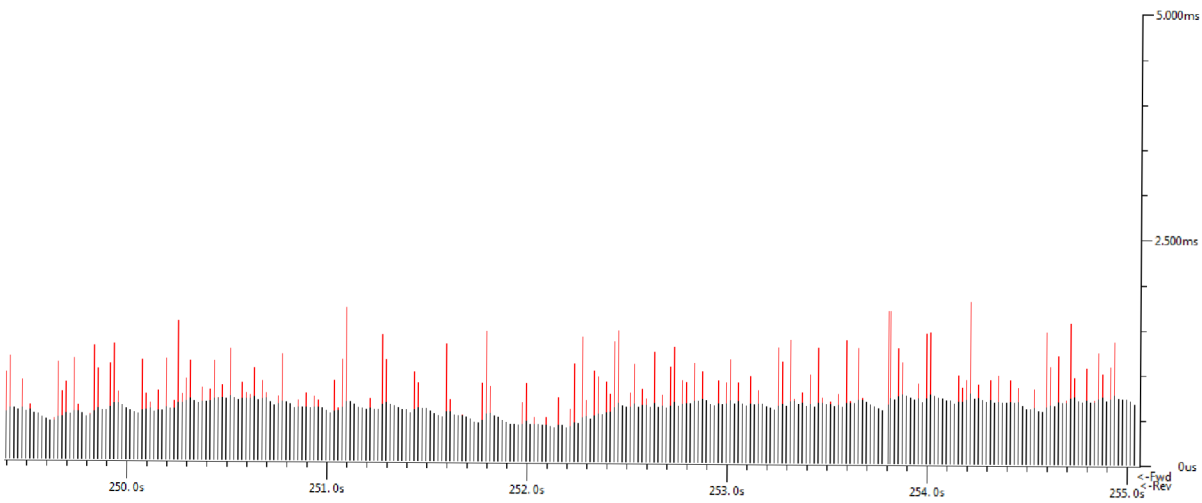


Figure 5.37: The uplink graph

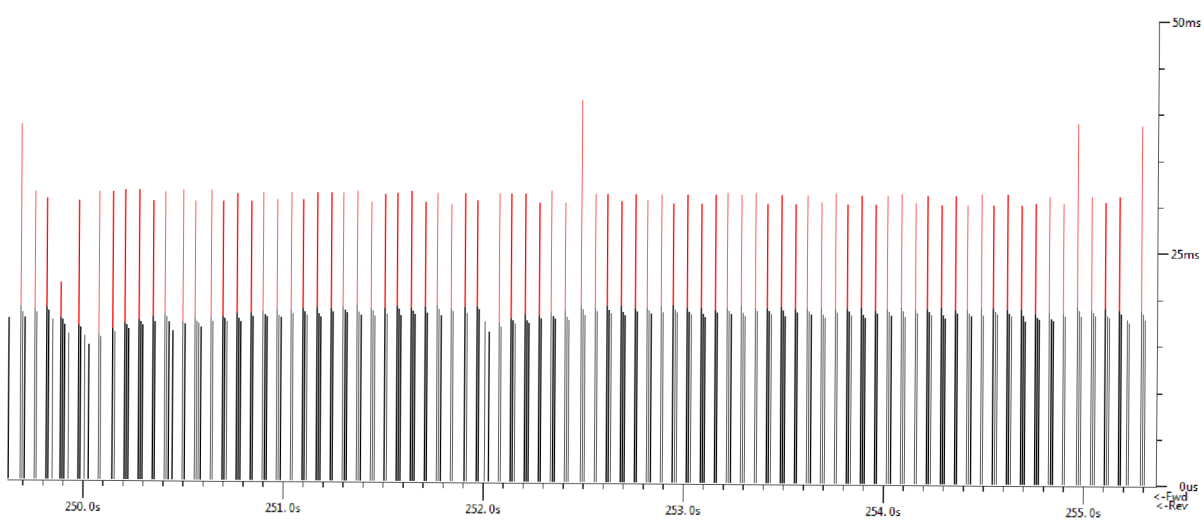


Figure 5.38: The downlink graph

Below in Figure 5.39 is the result obtained by the omnipeak:

Figure 5.39: The view of R-Factors for both streams from omnipeak.

5.2.4.1 Public connection without load (Low bit rate)

We connected the wire to port number 1, followed by decreasing the bit rate of the multiplexer (DSLAM). The multiplexer shows in Figure 5.40 no data flow, the two softwares were operated and the call was made. Included in Table 5.41 the values which we obtained are noted. The obtained two streams are in Figure 5.42 and the jitter graphs are in Figures 5.43 and Figure 5.44:

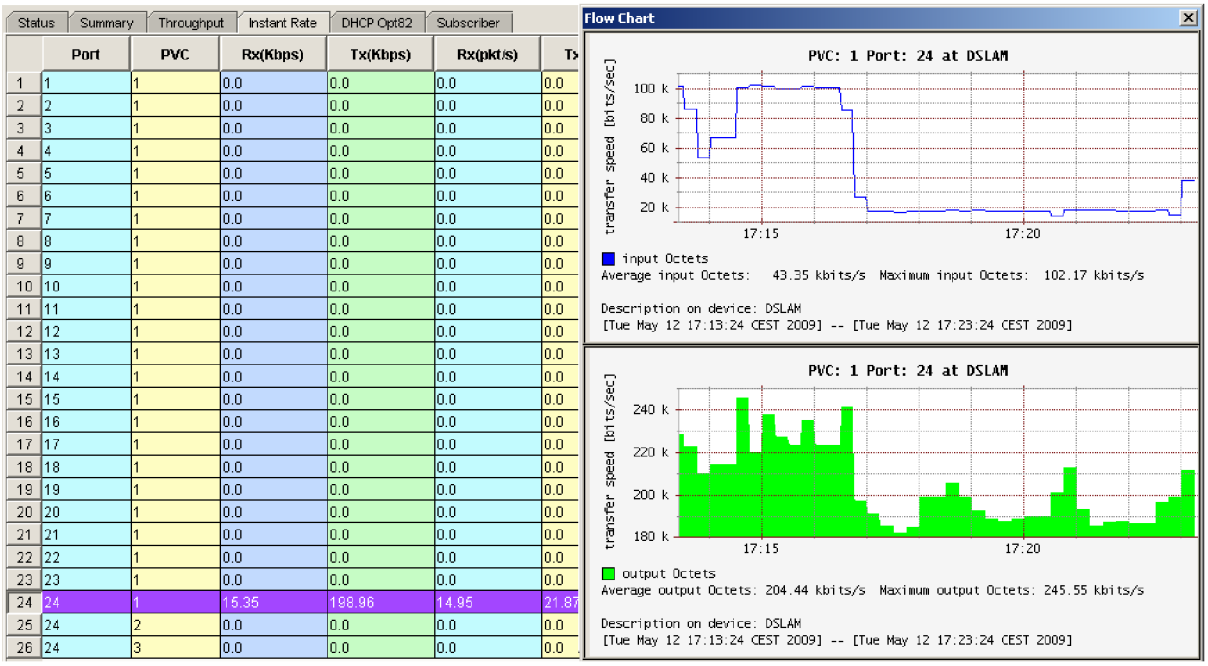


Figure 5.40: The usage of DISLAM.

300 Kbps	PUBLIC Low bit-rate			
	Chanel Released		Chanel Overloaded	
	upstream	Downstream	Upstream	downstream
Total packets	14088	14101	10955	10622
Packets lost	0	4	0	445
MAX delta (ms)	188.58	370.29	27.07	305.5
MAX Jitter (ms)	14.49	41.77	2.5	41.07
Mean jitter (ms)	0.61	18.57	1.74	22.34
R-Factor	92	18	92	11
CQ	(92-92)	(18-91)	(92-92)	(11-12)

Table 5.41: Public connection low bit rate with and without load.

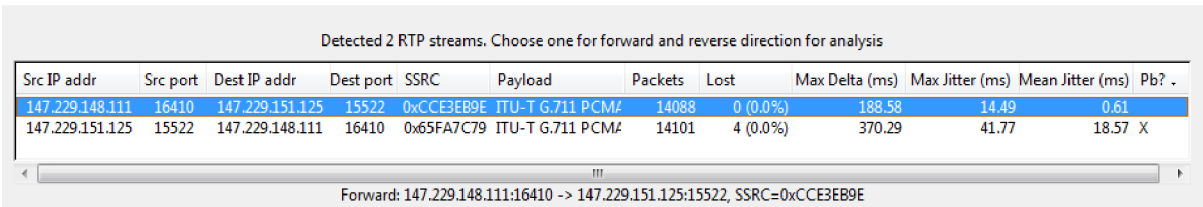


Figure 5.42: The two streams.

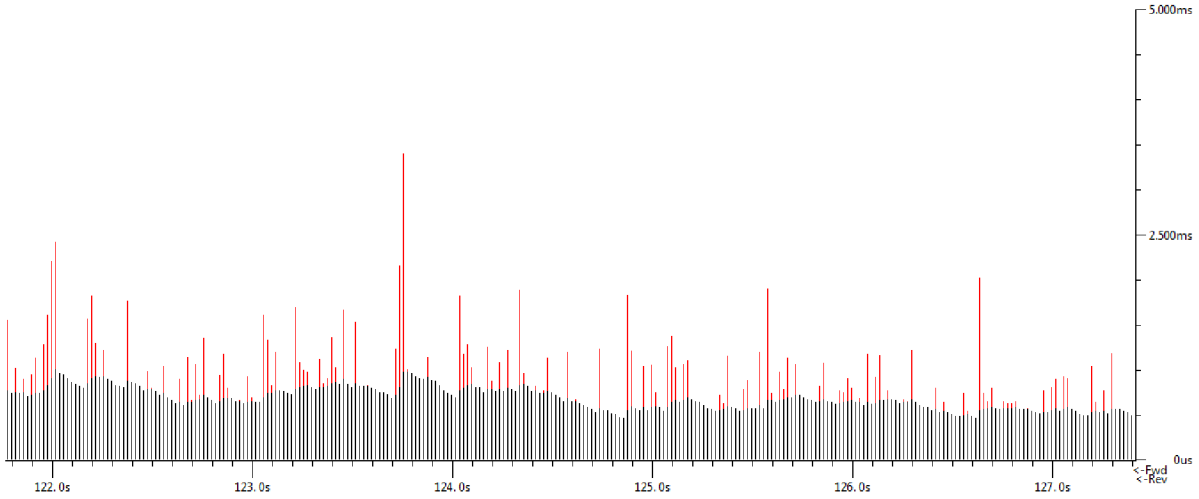


Figure 5.43: The uplink graph

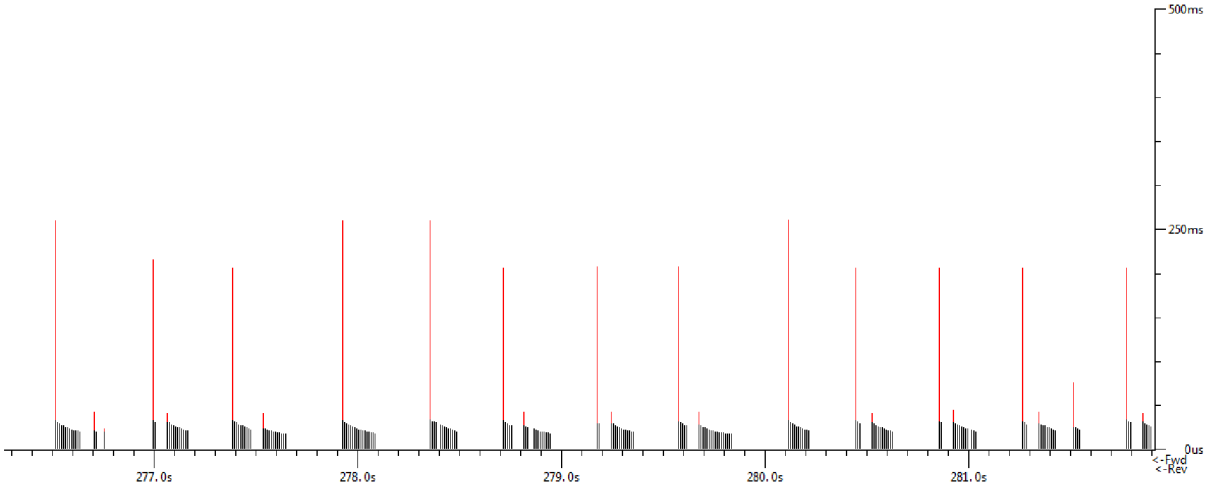


Figure 5.44: The downlink graph

Below in Figure 5.45 is the result obtained by the omnipeak:

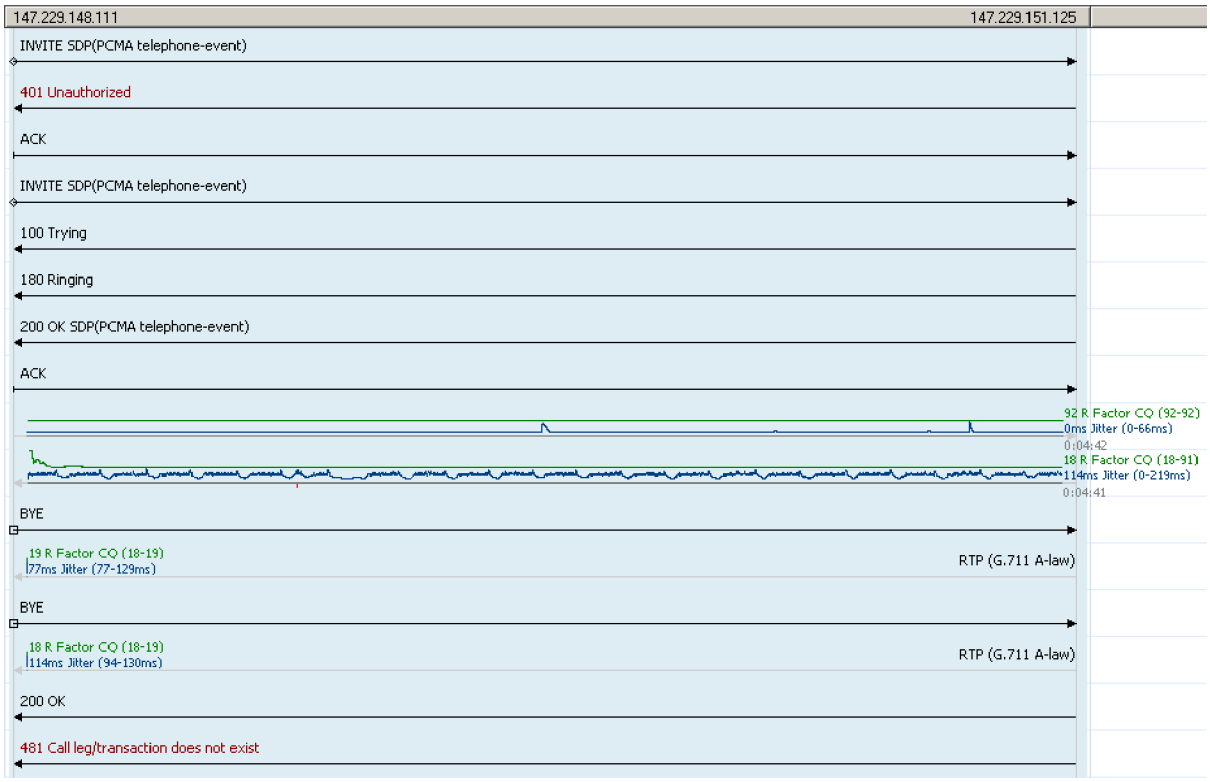
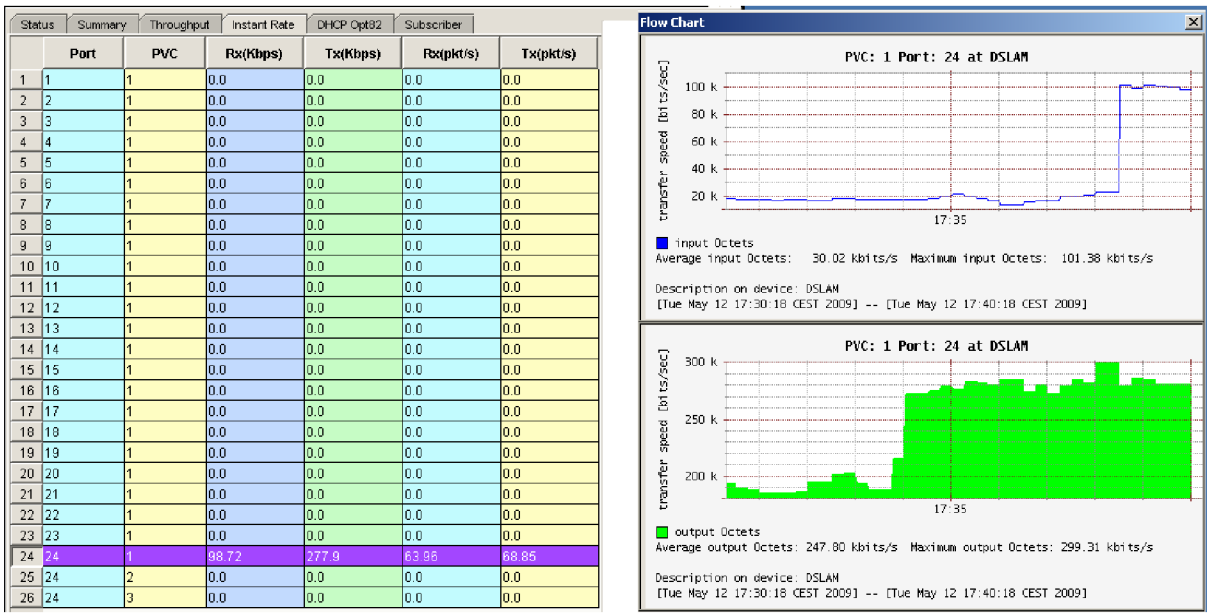


Figure 5.45: The view of R-Factors for both streams from omnipeak.

5.2.4.2 Public Connection with load (Low bit rate)

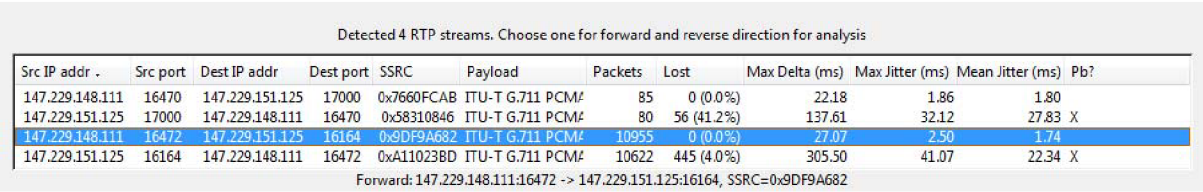
In this case we upload data from the PC which is connected to port number 4, and then we added the values to the previous Table 5.41 to compare the results. The obtained two streams are in Figure 5.47 and the jitter graphs are in Figures 5.48 and Figure 5.49:



Status | Summary | Throughput | Instant Rate | DHCP Opt82 | Subscriber

	Port	PVC	Rx(Kbps)	Tx(Kbps)	Rx(pkt/s)	Tx(pkt/s)
1	1	1	0.0	0.0	0.0	0.0
2	2	1	0.0	0.0	0.0	0.0
3	3	1	0.0	0.0	0.0	0.0
4	4	1	0.0	0.0	0.0	0.0
5	5	1	0.0	0.0	0.0	0.0
6	6	1	0.0	0.0	0.0	0.0
7	7	1	0.0	0.0	0.0	0.0
8	8	1	0.0	0.0	0.0	0.0
9	9	1	0.0	0.0	0.0	0.0
10	10	1	0.0	0.0	0.0	0.0
11	11	1	0.0	0.0	0.0	0.0
12	12	1	0.0	0.0	0.0	0.0
13	13	1	0.0	0.0	0.0	0.0
14	14	1	0.0	0.0	0.0	0.0
15	15	1	0.0	0.0	0.0	0.0
16	16	1	0.0	0.0	0.0	0.0
17	17	1	0.0	0.0	0.0	0.0
18	18	1	0.0	0.0	0.0	0.0
19	19	1	0.0	0.0	0.0	0.0
20	20	1	0.0	0.0	0.0	0.0
21	21	1	0.0	0.0	0.0	0.0
22	22	1	0.0	0.0	0.0	0.0
23	23	1	0.0	0.0	0.0	0.0
24	24	1	98.72	277.9	63.98	68.85
25	24	2	0.0	0.0	0.0	0.0
26	24	3	0.0	0.0	0.0	0.0

Figure 5.46: The usage of DSLAM



Detected 4 RTP streams. Choose one for forward and reverse direction for analysis

Src IP addr	Src port	Dest IP addr	Dest port	SSRC	Payload	Packets	Lost	Max Delta (ms)	Max Jitter (ms)	Mean Jitter (ms)	Pb?
147.229.148.111	16470	147.229.151.125	17000	0x7660FCAB	ITU-T G.711 PCMA	85	0 (0.0%)	22.18	1.86	1.80	
147.229.151.125	17000	147.229.148.111	16470	0x58310846	ITU-T G.711 PCMA	80	56 (41.2%)	137.61	32.12	27.83	X
147.229.148.111	16472	147.229.151.125	16164	0x9DF9A682	ITU-T G.711 PCMA	10955	0 (0.0%)	27.07	2.50	1.74	
147.229.151.125	16164	147.229.148.111	16472	0xA11023BD	ITU-T G.711 PCMA	10622	445 (4.0%)	305.50	41.07	22.34	X

Forward: 147.229.148.111:16472 -> 147.229.151.125:16164, SSRC=0x9DF9A682

Figure 5.47: The two streams

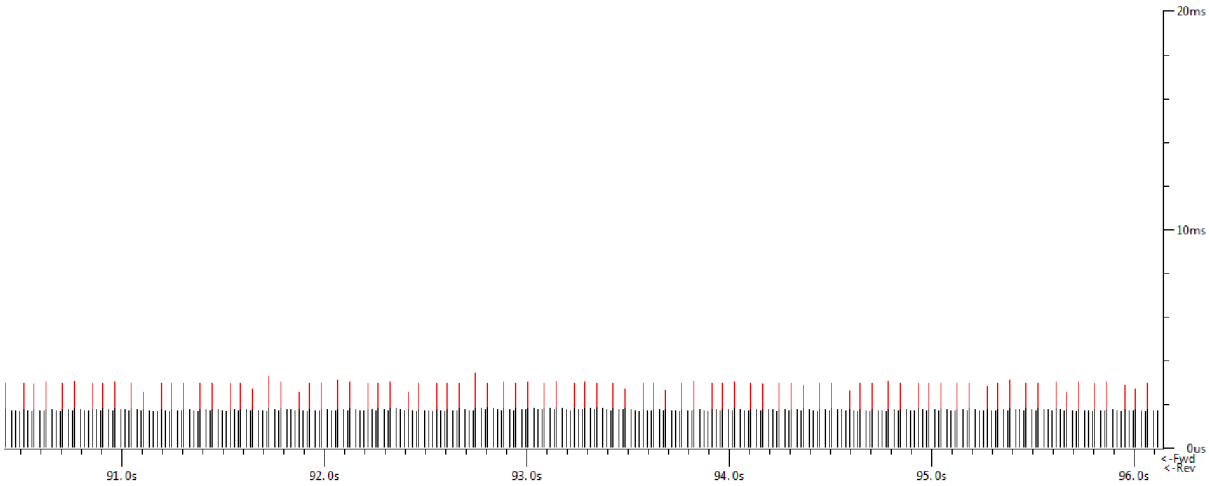


Figure 5.48: The uplink graph

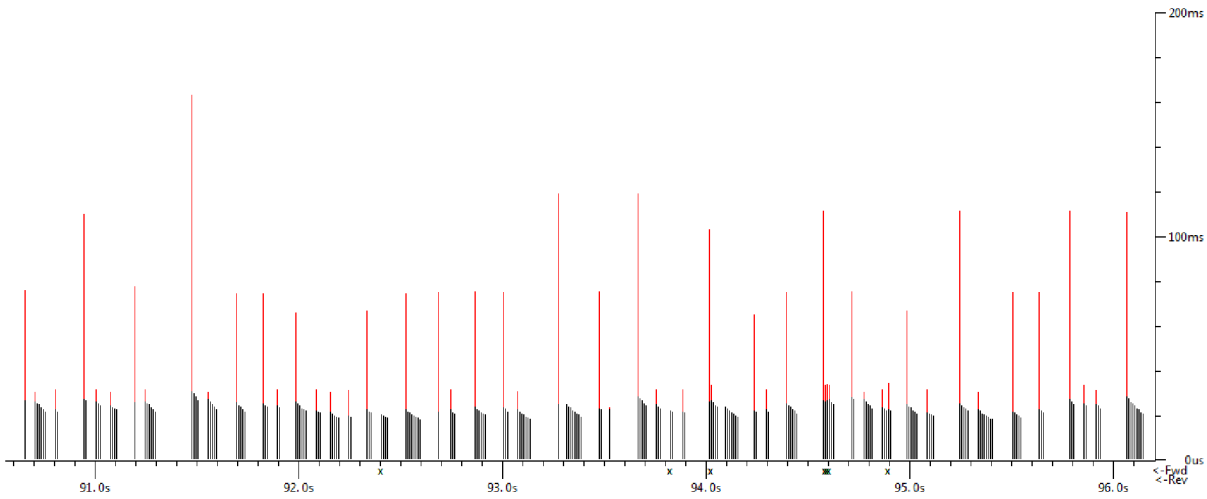


Figure 5.49: The downlink graph

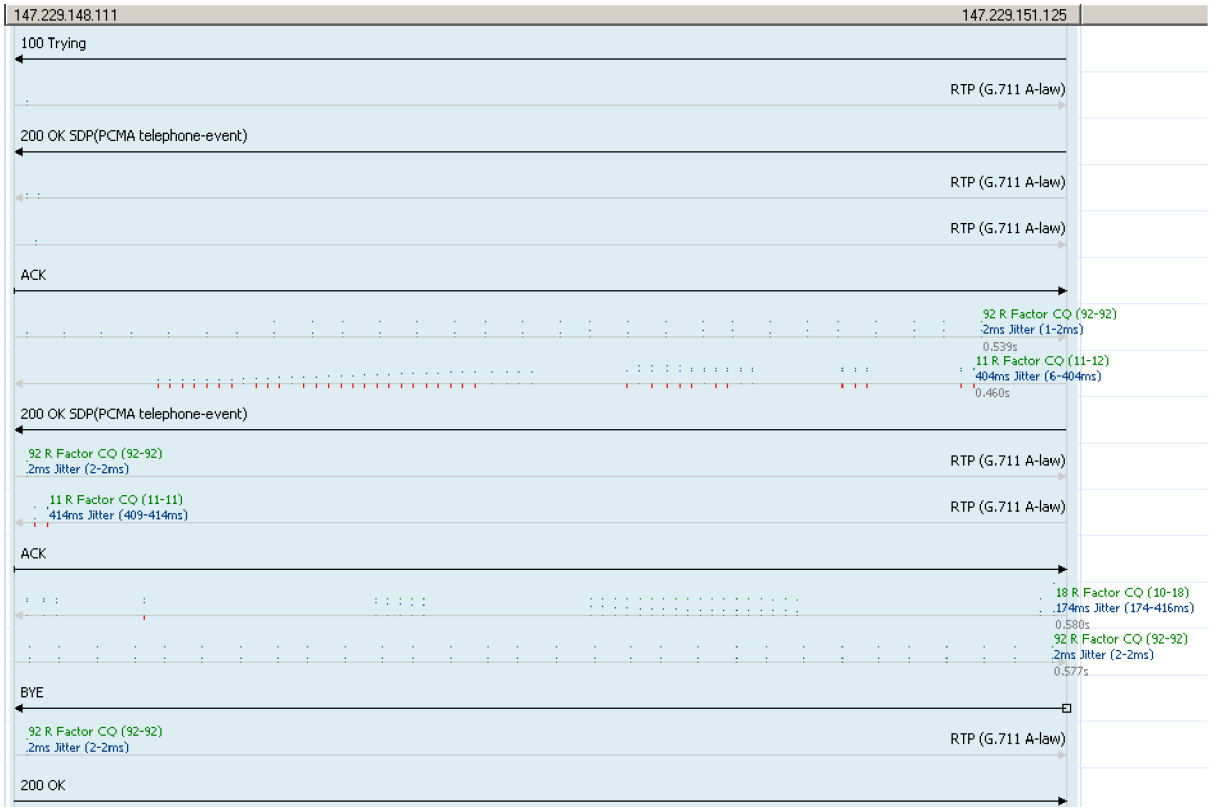


Figure 5.50: The view of R-Factors for both streams from omnipeak

5.2.5 VLAN Connection with and without load (various bit rate)

We connected the wire to port number 2, we measured the bit rate at (628 – 948 -1200) Kbps along with the difference between each stage, with and without load. We obtained the values and we displayed the result in Tables 5.51, 5.52 and 5.53.

628 Kbps	Chanel Released		Chanel Overloaded	
	Up	Down	up	Down
jitter	1 (0-1)	85 (0-185)	0 (0-1)	60 (0-177)
R-Factor	92	84	92	78
CQ	(92-92)	(63-92)	(92-92)	(64-92)

Table 5.51: Results at 628 Kbps.

948 Kbps	Chanel Released		Chanel Overloaded	
	Up	Down	up	Down
jitter	0 (0-98)	30 (0-204)	0 (0-1)	41 (0-181)
R-Factor	92	87	92	86
CQ	(91-92)	(85-92)	(91-92)	(86-92)

Table 5.52: Results at 948 Kbps.

1.2 Mbps	Chanel Released		Chanel Overloaded	
	Up	Down	up	Down
jitter	0 (0-1)	1 (0-187)	0 (0-1)	7 (0-172)
R-Factor	92	85	92	74
CQ	(92-92)	(81-92)	(92-92)	(70-92)

Table 5.53: Results at 1.2 Mbps.

5.2.6 PUBLIC connection from far distance (High bit rate)

In the final step for a public connection, we made an external call (distant from the laboratory) to check out if the results coincide with the other results that we obtained from the public connection which is established in the laboratory. The results are shown in Table 5.54 and the graphical representation of this result is shown in Figure 5.55.Where the gaps are displaying that the call had disconnected.

	Clear	
	Up	Down
jitter	0 (0-10)	14 (0- 7 635)
R-Factor	88	65
CQ	(77-92)	(42-92)

Table 5.54: Public connection with an external call.

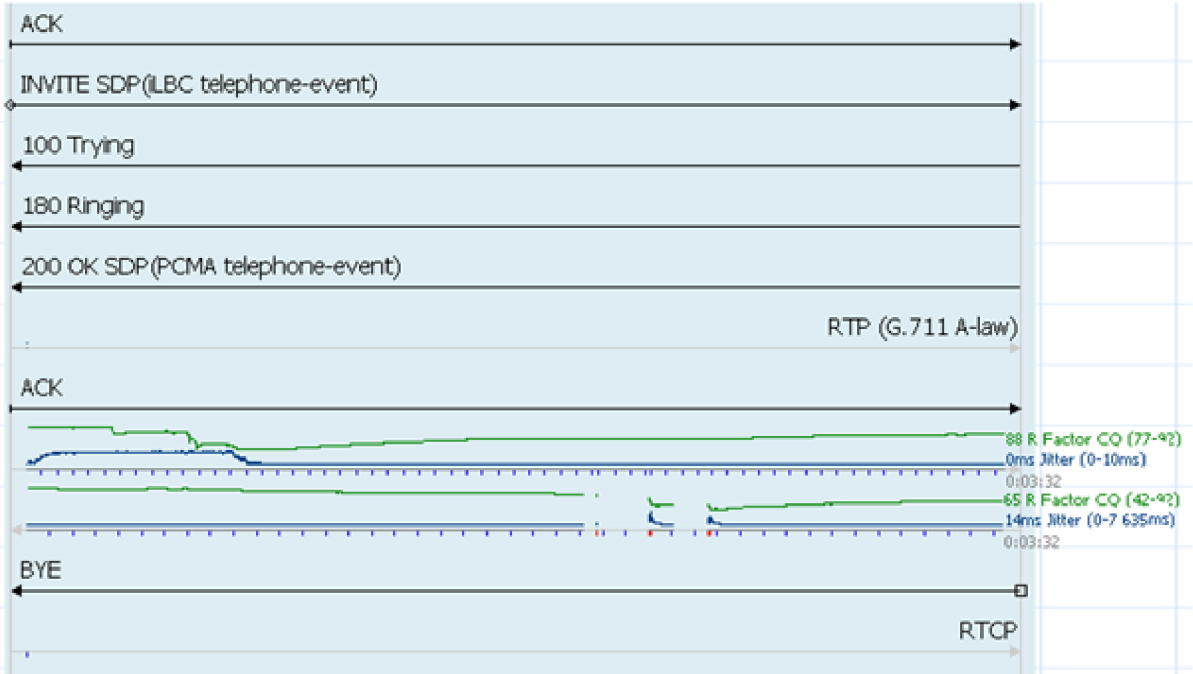


Figure 5.55: The view of R-Factors for both streams from omnipeak.

We have combined all the results obtained in the following table below:

VLAN 303 Without overload

BW	314 (Kbps)		628 (Kbps)		948 (Kbps)		1.2 (Mbps)		7.6 (Mbps)	
stream	uplink	downlink	uplink	downlink	uplink	downlink	uplink	downlink	uplink	downlink
jitter(ms)	0 (0-1)	1(0-162)	1 (0-1)	85 (0-185)	0 (0-98)	30 (0-204)	0 (0-1)	1 (0-187)	0 (0-32)	0 (0-192)
R-Factor	92	91	92	84	92	87	92	85	92	89
CQ	(92-92)	(91-92)	(92-92)	(63-92)	(91-92)	(85-92)	(92-92)	(81-92)	(92-92)	(89-92)

VLAN 303 With overload

BW	314 (Kbps)		628 (Kbps)		948 (Kbps)		1.2 (Mbps)		7.6 (Mbps)	
stream	uplink	downlink	uplink	downlink	uplink	downlink	uplink	downlink	uplink	downlink
jitter (ms)	0(0-126)	20(0-161)	0(0-1)	60 (0-177)	0(0-1)	41(0-181)	0(0-1)	7(0-172)	0(0-32)	1 (0-191)
R-Factor	92	90	92	78	92	86	92	74	92	89
CQ	(92-92)	(81-92)	(92-92)	(64-92)	(91-92)	(86-92)	(92-92)	(70-92)	(92-92)	(89-92)

Public Without overload

BW	314 (Kbps)		628 (Kbps)		948 (Kbps)		7.6 (Mbps)		7.6 (Mbps)	
stream	uplink	downlink	uplink	downlink	uplink	downlink	uplink	downlink	uplink	downlink
jitter (ms)	0 (0-66)	114 (0-219)	-	-	-	-	-	-	2 (0-2)	2 (0-199)
R-Factor	18	92	-	-	-	-	-	-	90	92
CQ	(18-91)	(92-92)	-	-	-	-	-	-	(89-92)	(92-92)

Public With overload

BW	314 (Kbps)		628 (Kbps)		948 (Kbps)		7.6 (Mbps)		7.6 (Mbps)	
Stream	uplink	downlink	uplink	downlink	uplink	downlink	uplink	downlink	uplink	downlink
jitter (ms)	2 (1-2)	404 (6-404)	-	-	-	-	-	-	0 (0-14)	39 (0-198)
R-Factor	11	92	-	-	-	-	-	-	88	92
CQ	(11-12)	(92-92)	-	-	-	-	-	-	(86-92)	(92-92)

6. Conclusion

The quality of the voice is depending upon the bit rate and the intermediate Switches, each one of those are involved with each other. For example if we are having a low bit rate of the multiplexer in the other case we should have constant streaming intermediate Switches to obtain better communications and vice versa.

The voice can be transmitted in a small channel which can be integrated with the data, whereas transmitting a video cannot be integrated with the data unless it is through a high transmission data process which means high transmitting data is

required and payload to offer a good quality. ATM is providing a good quality of service, and the reason is the cells are driven along the same restricted path which prevents the cells to divert from the path to required destination.

In the laboratory we have used four ports of the ADSL modem as following: The first port is used for the public connection without the concept of the triple play connection. The second one is used for the VLAN to the VOIP with the concept of the triple play principle to analyze the voice and integrate it with internet data. As for the third one it's connected to the VLAN for the IPTV again with the concept of the triple play principle.

We have examined the data transmission through two cases: in the first case, we used the maximum bandwidth efficiency, which means, we have downloaded the data in such way to reach the maximum bandwidth efficiency of the DSLAM through downloading as much data as the bandwidth could afford, while in the second case, we used the minimum bandwidth efficiency, and we have changed the speed of the data transmission from the maximum value to the minimum value in several steps

From the results of the lab exercise we realized that the concept of the Triple play connection is more qualified in the speed of data transmission than a regular Public connection speed at the rate of 314 Kbps. We had noticed that no significant difference existed between both connections at the speed of 7.6 Mbps, since both connections operated with high quality.

Therefore we took a further step to emphasize whether the analysis was accurate or not. We established an actual public connection to test voice quality outside the university campus. Subsequent to the testing we discovered that the R-Factor of the Public connection specified a lower level in quality of approximately 60% than the obtained results within the connection which we created in the university network.

Glossary

ADSL
asymmetric digital subscriber

ATM
asynchronous transfer mode

CAP
Carrierless Amplitude and Phase

CATV
Community Antenna Television

CLEC
Competitive local exchange carrier

CPE
Customer-premises equipment

DMT
Discrete Multi-tone

DSL
digital subscriber line

DSLAM
Digital Subscriber Line Access Multiplexer

DVD
digital versatile disc

E1
E-carrier Level 1

EOC
Embedded Operations Channel

GFC
Generic Flow Control

HDTV
High Definition Television

HEC
Header Error Control

ILEC
Incumbent local exchange carrier

IP
Internet protocol

IPTV
Internet Protocol Television

ISDN
integrated services digital network

LAN
local area network

LANE
local area network Emulation

LLC
Logical Link Control

MMF
multimode fiber

MPEG
Moving Pictures Experts Group

MPLS
Multiprotocol Label Switching

MPOA
Multiprotocol over ATM

MTU
Maximum Transmission Unit

NAT
Network address translation

NNI
Network to Network Interface

OUI
Organizationally Unique Identifier

PID
Protocol Identifier

POTS
plain old telephone service

PPP
Point to Point Protocol

PSD
Power Spectral Density

PT
Payload Type

QoS
quality of service

RSVP
Resource Reservation Setup Protocol

RTCP
Real time control protocol

RTP
Real-Time Transport Protocol

SDH
Synchronous optical networking

SIP
Session Initiation Protocol

SNAP
Subnetwork Access Protocol

SONET
synchronous optical network

STP
shielded twisted-pair

T1
Transmission Level 1

TCP
transmission control protocol

TTY
Teletypewriter

UBR
unspecified bit rate

UNI
User to network interface

VCC
Virtual Channel Connection

VCI
Virtual channel identifier

VCI
Virtual Channel Identifier

VLAN
Virtual local area network

VoIP
Voice over Internet Protocol

VOD
Video on demand

VPI
Virtual Path Identifier

6. References

[1] GINSBURG, David. Implementing ADSL. AWL, Inc., Massachusetts USA. November 1999, ISBN 0-201-657 60-0.

[2] SUMMERS, Charles K. ADSL Standards, Implementation and Architecture. CRC Press LLC, 200FCorporate Blvd., N.W., Boca Raton, Florida 1999,ISBN 0-8493-955-X.

[3] Detken , K.-O.: ATM in TCP/IP environment: Adaptation and Effectiveness; ACTS PROJECT AC094 EXPERT: ATM Traffic Symposium; Mykonos, Greece, 17-18 September 1997

[4] Sacket, G. C.; Metz, C. Y.: ATM and Multiprotocol Networking; McGraw- Hill series on Computer Communications; New York 1996.

[5] Yankee Group, Broadband Access Technologies, 2004. Telco Take on Cable with Video Delivery, Available: http://resources.bnet.com/topic/broadband+and+networking+and+the+yankee+group.html

[6] O'Shea, D., DSL decision time, 2005. New York, USA, Available: http://www.telephonyonline.com/mag/telecom_dsl_decision_time/index.html

[7] Westfall, R., Digital Loop Carrier, 2004 Available: https://bkr.bls.com/redirRenderDoc.jsp?book_id1015341088

[8]. Intel. H.264 & IPTV Over DSL, 2004, Available: http://www.envivio.com/pdf/whitepaper_H264_IPTV_Over_DSL.pdf

[9]. Internetworking Technology Handbook Cisco system, Available: http://www.cisco.com/en/US/docs/internetworking/technology/handbook/ito_doc.html

[10]. San Jose, Traffic Management for Testing Triple-Play Services, 2006 Available: http://www.altera.com/literature/wp/wp-trffc.pdf

[11]. International Engineering Consortium, Available: http://www.iec.org/online/tutorials/

[12] Alon beizer & Alon shulchami , December 1996 , Available: http://telem.openu.ac.il/students/courses/c20398/atmproj/home.htm

[13] John M. Cioffi et al. (2008), Scholarpedia, 3(8):3995. Available: http://www.scholarpedia.org/article/DSL

[14] TCP/IP Tutorial and Technical Overview December 2006 Available**:** http://www.redbooks.ibm.com/redbooks/pdfs/gg243376.pdf

[15] Abdulrahman Yarali, Internet Protocol Television (IPTV), Kentucky, USA.

[16]. Elander, L. (2005). New Opportunities for Cable with IPTV. (On-line), Available: http//:www.microsoft.com/tv/content/press/Oportunuities_for_Cable_with IPTV.mspx.

[17]. TDG Research. (2005). Global IPTV Revenue Will Top $17 Billion by 2010, Available: http//:www.emediawire.com/printer.php?prid=198746

[18] Paul Drew and Chris Gallon, Next-Generation VoIP Network Architecture, March, 2003, Fremont, CA, USA